An Efficient Method for Iterative Tolerance Design Using Monte Carlo Simulation.

A Thesis

Presented to the

Department of Mechanical Engineering

Brigham Young University

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

By

David V. Larsen

April 1989

This thesis, by David V. Larsen, is accepted in its present form by the Department of Mechanical Engineering of Brigham Young University as satisfying the thesis requirements for the degree of Master of Science.

Kenneth W. Chase

Kenneth W. Chase, Committee Chairman

Carl D. Sorensen

Carl D. Sorensen, Committee Member

1-13-89

Date

John C. Cannon

John C. Cannon, Graduate Coordinator

AKNOWLEDGEMENTS

I would primarily like to express gratitude to my wife and family for their support an patience with me while pursuing my graduate degree. My wife gave me the strength and help in every way she could without complaint. Her contributions to this work should not go unnoticed.

Secondly, I would like to thank Dr. Kenneth W. Chase and Dr. Carl D. Sorensen for their knowledge, leadership and patience. Their examples have inspired me in my academic pursuits as well as many others areas.

Finally, I would like to thank the Church of Jesus Christ of Later Day Saints and the department of Mechanical Engineering for developing an environment of leaning and integrity essential in the development of integrity and moral strength .

An Efficient Method for Iterative Tolerance Design Using Monte Carlo Simulation.

David V. Larsen

Department of Mechanical Engineering

M. S. Degree, April 1989

ABSTRACT

A new tolerance design method has been developed using an efficient hybrid of Monte Carlo simulation. This Method, called Optimal Design Simulation (ODS) , allows greater flexibility in modeling mechanical assemblies and allocating tolerances. The ODS method uses a technique of storing the compact assembly distribution characteristics for use with iterative design. This method also allows the use of non-Normal component distributions and non-linear assembly functions.

Traditional Monte Carlo and the ODS method are compared and evaluated. Both methods are then tested for accuracy in modeling a real system. Non-Normal component effects on an assembly as well as trends due to sample size and the number of components in an assembly are determined and discussed. Finally, the ODS method is used to assign tolerances for a real system based on a specified acceptance fraction. These results are compared to the known results for accuracy.

COMMITTEE APPROVAL:

Kenneth W. Chase

Kenneth W. Case, Committee Chairman

Carl D. Sorensen

Carl D. Sorensen, Committee Member

John N. Cannon

John N. Cannon, Graduate Coordinator

Table Of Contents

List of Figures

List of Symbols

Symbol	Description
α	Percentage of tolerance range.
β_1	Skewness.
β_2	Kurtosis.
λ_{1-4}	Generalized Lambda shape parameters.
θ	Assembly Dimension.
θ'	Initial estimation for θ.
σ	Standard deviation.
C_k	Coefficients required for the kth moments in ODS.
D	Half band width of the error band.
Lam_{1-4}	Lambda shape parameters.
M	Number of components in the assembly.
M_k	The kth moment of a distribution.
N	Sample size.
p	Probability.
T_i	Tolerance range of ith component.
$X_{(p)}$	Percentile function.
X_i	Specific dimension for the ith component.
$\overline{X}$	Average dimension of the components for the total simulation.
Y_i	Specific dimension for the ith assembly.
$\overline{Y}$	Average dimension of the assembly for the total simulation.
Z	Number of standard deviations.

List of Tables

1. Introduction

1.1 Tolerance Analysis Today

Design and production of manufactured products has become extremely competitive in American and International markets. A type of panic has struck American industry due to a significant drop in the use of American engineered and manufactured products. To compete in today's international market, companies are being forced to produce higher quality products at lower prices. Much time and money is being invested to find new technologies that will allow America to regain lost ground in the world market.

Many engineers try to design high quality products simply by making the part specifications more restrictive. These stringent specifications are often the cause of conflict between the designer and manufacturer. If a tight tolerance is desired then the designer must take into account the high cost of precise manufacturing techniques. On the other hand, if the designer is solely concerned about manufacturing costs and requests less restrictive tolerances, the components may not assemble or the assembly may perform poorly resulting in costly scraping, re-work or customer dissatisfaction. Designers are often insensitive to the difficulties involved with real manufacturing processes, while manufactures do not consider all design issues. The objective of the engineer is to properly assign component tolerances in such a way as to reduce costs and increase quality. To aid the engineer in his decision making process tolerance analysis methods were developed.

Tolerance analysis can be defined as the estimation of dimension propagation in a mechanical assembly due to variation in the manufactured component dimension. This type of analysis can be a valuable tool in the effort to reduce manufacturing costs and improve efficiency. By using tolerance analysis the designer assure that critical assembly clearances will be met in spite of manufacturing variations. Tolerance analysis also creates a common link for the designer and manufacturer to communicate and improve the overall production process.

There are several methods of tolerance analysis being used by designers. Two of the most common are Worst Limits and Root Sum Squared (statistical). These elementary methods are conceptually simple and time efficient. However, in many cases these common methods are an inadequate approach to product development. Worst Case analysis is based on the assumption that all parts in an assembly are simultaneously at their high or low limits. The assembly dimensions are found by evaluating all components at their tolerance limits and adding them linearly (see equation 1.1). Selecting design tolerances based on these conditions results in unnecessarily tight tolerances and high production costs. In reality, since the worst case has such a low probability of occurrence, tolerances may be relaxed, reducing production costs.

$$T_{Asm} = \sum T_i \qquad \text{Equation 1.1}$$

Root Sum Squared analysis uses statistical theory to analyze tolerance interaction. The Root Sum Squares (Statistical) does not classify the tolerances as a hard physical limit but as an indication of the standard deviation or variance. Statistics teaches that the sum of two or more independent variables' standard deviations is the standard deviation of the dependent variable [17]. Root Sum Squared analysis is based on this basic statistical concept and is represented by equation 1.2. In this case, the assembly dimensions are not found through a linear summation but as the square root of the sum of the squares of the component tolerances.

$$T_{Asm} = \sqrt{\sum T_i^2} \qquad \text{Equation 1.2}$$

There are several limitations in using the statistical method. It cannot accurately predict the yield of assemblies with low part counts, small batch sizes or which have non-Normal or biased distributions of component dimensions. It is also limited to linearized assembly equations and the assumption that the resultant assembly has a Normal or Gaussian distribution [20].

Greenwood [8] proposed a modified Root Sum Squared method which uses a Normal distribution with a mean shift to account for biased distributions. This method is a better approximation for many processes but it is still inadequate for highly non-Normal distributions and nonlinear assembly functions. Because of the weaknesses in these techniques, more complicated methods of tolerance analysis must be used to achieve the necessary accuracy needed to produce higher quality parts at lower costs.

Advanced tolerance analysis techniques offer models with a higher degree of accuracy for these difficult non-Normal, non-linear cases. These advanced methods permit the setting of realistic tolerances that can assure proper performance at a minimum production cost. The advanced techniques that will be discussed in this thesis are the Method of System Moments, and Monte Carlo Simulation.

The Method of System Moments uses the first four moments of the component distributions to determine the final assembly distribution. It is an efficient method, but the expressions for the moments of the resultant distribution in terms of the component moments is very complex.

Monte Carlo simulation is a conceptually simple computer tool which allows us to mathematically approximate the manufacturing and assembly process, and make design decisions based on the performance of these processes thereby providing the opportunity to analyze the design before production. This simulation method uses random variates to generate numerous sets of components with small variations in their dimensions. These component dimensions are combined through the use of an assembly function to create the assembly distribution. The resultant distribution generated by both Monte Carlo, and the Method of System Moments is then used to calculate the percentage of rejected assemblies in the total batch. Evans [6] presents a comprehensive survey of these methods.

Both the Method of System Moments and Monte Carlo require that statistical distributions be assigned as a representation of component variation, however, the prediction of the rejection fraction is handled differently in the two cases. The Method of System Moments uses the first four statistical moments of the components to determine the parameters of the resultant or assembly distribution. The rejection rate can then be calculated by numerical integration of the resultant distribution or from tables. The Method of System Moments will be discussed in greater detail in Chapter 5. The rejection fraction for Monte Carlo is calculated by physically counting the simulated assemblies that fall outside of the pre-specified tolerance ranges and then dividing by the number of total simulations.

The advanced methods are more accurate than the elementary methods but are often difficult to use effectively as analysis tools since an accurate estimate of each component's statistical distribution must be determined [8]. The component distributions are difficult to determining during production and can only be roughly estimated in the design stage. This often results in erroneous assumptions and poor designs. Therefore, in order to have an accurate analysis, a precise method of characterizing various manufacturing processes must be found. For example, turning, milling and drilling all have different characteristic distributions describing the manufacturing process. To model these processes correctly, the characteristic distribution must be found along with typical batch-to-batch variation in mean shift, set up error, etc.

Another drawback to the advanced methods, particularly Monte Carlo, is the extensive computer usage. To attain acceptable accuracy at low rejection rates using the Monte Carlo method, a simulation of 100,000 or more assemblies is needed [8]. This effectively eliminates the use of classical Monte Carlo from any type of iterative design work.

1.2 Problem Statement

The purpose of this thesis is to develop practical tolerance analysis methods based on advanced statistical theory. The specific objective is to improve the efficiency of Monte Carlo simulation making it more suitable for design iteration and for tolerance allocation applications.

If statistical simulations can be used more effectively and economically for tolerance specification, the overall manufacturing process could be greatly improved. The cost of re-work and part rejection could be effectively eliminated or reduced to an acceptable level before the design reaches the manufacturer.

1.3 Procedure

The scope of this proposed research project covers five major areas:

1) Development of a procedure for generating random variates for Normal and non-Normal distributions covering the full range of skewness and kurtosis values required to describe common manufacturing processes.

2) Development of a traditional statistical simulation tool for tolerance analysis as a benchmark for comparison purposes.

3) Development of a Hybrid method of simulation which combines Monte Carlo with the Method of System Moments. This was used for comparison tests and as a framework for the proposed design method.

4) Development of a new modified Hybrid method called the Optimal Design Simulation Technique. This method stores the first set of

variates in non-dimensional form and retains the tolerances as design variables.

5) Evaluation of the feasibility of iterative tolerance allocation using the Optimal Design Simulation technique.

6) Comparision and evaluation of the three previously mentioned methods of tolerance simulation and design.

1.4 Delimitations

The limitations of this research were:

1) Allow up to 40 components in an assembly, but only 20 free re-allocatable design tolerances.

2) The non-Normal distributions were approximated by the Generalized Lamda distribution only.

3) The research was limited to linear assemblies, but will include sensitivities for future non-linear applications.

2. Current Work in Tolerance Analysis

Most of the tolerance analysis being done today with statistical theory and is purely analysis. Rarely is any design work being done. This is due to the difficulties involved in conceptually understanding statistical theory and the extensive amount of computer resources needed to accurately simulate a real system. For example to use traditional Monte Carlo simulation in an iterative design process is extremely expensive in terms of human and computer time. However, to achieve the accuracy needed for a good design, Monte Carlo is the best choice. Therefore, to use this method in an iterative design environment, some modifications must be performed.

2.1 Current Publications

Many works have been compiled to introduce the theory of basic statistics [18][19]. These works generally do not have the engineer in mind and generally do not examine any applications for the engineer. They do, however, identify most of the important concepts necessary in the development of engineering applications and must be studied to ensure correct application of statistical principles in the engineering environment.

Samuel S. Shapiro and Alan J. Gross [16] present a good introduction to the basic concepts involved with statistical modeling of physical systems. They also explain in detail two statistical techniques, Monte Carlo simulation and the Method of System Moments, working examples for each method. Shapiro and Gross make an important contribution to this thesis with the introduction of the Generalized Lambda family of distributions developed by

Ramberg, Dudewicz, Tadikamalla and Mykytka [14]. Several methods of variate generation are presented, however, the Generalized Lambda family appears to be the best for the purposes of this research. While Shapiro and Gross do an excellent job in introducing statistical modeling and variate generation, no evaluation or application of these methods to tolerance analysis is discussed.

There have been in depth studies on the specific uses of the most common advances statistical methods such as Monte Carlo simulation [15][21] and the Method of System Moments [4]. Generally, research is done on the various aspects in Monte Carlo methods and their application to the simulation of physical systems. Discrete methods of calculating the mean and variation are introduced and tested for accuracy. Many variations of the basic Monte Carlo are also introduced and discussed. Rubinstein [21] introduce the concept of assembly function linearization through partial derivatives (sensitivity). He also discuses the idea of Monte Carlo optimization. This is as close as any of the information covered came to suggesting the using of Monte Carlo for design purposes.

Gerald J. Hahn and Samuel S. Shapiro [9] introduce some basic statistical modeling techniques specifically for the engineer. The main thrust of this book is to enlighten the engineer to statistical methods. They develop the idea of product performance based on manufacturing fluctuation and the application of this idea to practical engineering problems. The method of system moments and Monte Carlo simulation are also discussed. One of the strengths of the book is the comparison of the different distributions. Various distributions are classified and compared as to their functionality with modeling techniques in real engineering applications. An efficient graphical comparison is developed [9,pg.221] and used to determine the effectiveness of most well known distributions. This graphical approach quickly determines if a distribution is suitable for a desired situation. The lambda distribution is not covered in these comparisons.

William H. Greenwood [9] does an excellent job of presenting the specific uses for statistical methods in tolerance analysis. He evaluates the elementary methods currently being used in industry and concludes that they are inadequate. He also introduces a new method of analysis using a mean-shifted Normal distribution to more accurately represent real processes. Greenwood is concerned mainly with a new analysis technique and does not cover design issues directly.

Monte Carlo simulation was related directly to tolerance analysis by Corlew and Oakland [3]. In this paper tolerance analysis with Monte Carlo simulation is explained simply in a step-by-step fashion. This was a good introductory paper to statistical tolerance analysis. Cox [4] takes a different approach to tolerance analysis. He concentrates on the Method of System Moments as a tolerancing tool. He explains in some detail the method and advantages of this technique. Cox concedes in his conclusions that for non-linear assembly functions this method becomes very difficult.

Dr. Ken Chase and William H. Greenwood define the basic concerns and pitfalls of mechanical tolerance analysis [2]. In this paper they define the difference between tolerance analysis and tolerance allocation. They also discuss many of the benefits and disadvantages of the commonly used analysis methods as well as Greenwood's mean shift model. A promising new method of optimal cost vs. tolerance allocation is introduced, where tolerances are weighed in terms of the production costs.

The most comprehensive view of tolerance analysis is given by Evans [6]. In this paper an extensive history of tolerance analysis is presented. The point is made that the four most effective methods of tolerance analysis are: Stack Tolerancing, The Method of System Moments, The Quadrature Technique, and Monte Carlo simulation. He states that while Monte Carlo is a powerful tool, it should only be used when the other methods are limited by precision. In a good design the rejection rate should be low, leading to the necessity of a high degree of precision. Therefore, most analysts conclude that for most real manufacturing processes Monte Carlo simulation is the method of choice, however, it is frequently used incorrectly.

2.2 Available Software

There currently exists software that allows the engineer to perform general tolerance analysis. Most of these packages are based on Monte Carlo simulation. Assembly Variation Simulation System (AVSS) is a computer simulation program developed by Deere & Company to statistically analyze tolerance stack-up of complicated two-and three-dimensional assemblies [5][13]. This package also calculates the assembly sensitivity to part variation. Two real life problems faced by John Deere are presented and solved. Through the use of these statistical techniques. The main problem with the articles that describe this method is that they do not go into enough detail about the actual method used in the algorithm. They simply state that a problem existed and with the use of the AVSS software it was solved.

Grossman [7] presents a graphical simulation system for three-dimensional solid tolerancing. Procedural Geometric Modeling System (PGMS) allows the use of parameters, such as tolerances, to characterize arbitrary attributes of a part. This allows the user to specify tolerances on primitives such as cuboids, cones etc. With this method only the uniform and normal distributions are available for tolerance characterization, putting severe limits on the modeling technique.

TOLCON [11] is a software package used to generate Monte Carlo simulation for linear or non-linear assemblies. It uses traditional Monte Carlo simulation with an interactive user interface to define part dimensions, simulation batch size and statistical analysis. Since TOLCON uses traditional Monte Carlo simulation as its base, any type of iterative design process is feasible but not practical. This software package does a good job with the user interface, but the principal problem involved with Monte Carlo simulation is not discussed. This problem being the extensive computer time involved in the simulation of complex assemblies.

CATS.BYU is an interactive program for engineers and designers to assist them in the evaluation of tolerance specification. It assists the designer in the selection and allocation of tolerances for complex assemblies. It has four methods available to the user for tolerance specification including an on-line tolerance reference handbook, Worst case analysis, Statistical, and Allocation by Optimization. At present, CATS is limited to one-dimensional tolerance stacks, but two-and three-dimensional programs are under development, along with interactive CAD modeling interfaces.

3. Random Variate Generation

3.1 Basic Variate Generation

The first step of this research was the development of the random number generating algorithms. The fundamental component of any accurate simulation package is the random number generator. If the random number generation introduces any error or bias, the more sensitive analysis methods will incorporate this error into the model resulting in an inaccurate analysis. Since it is necessary to simulate various types of manufacturing processes, a method of generating a wide selection of distribution shapes must be determined. The basic method of variate generation is to start with a uniform random number generator that produces values from zero to one. These uniform values are used as percentiles and mapped to the desired distribution's cumulative function. The various variant generating algorithms used in this research were developed and tested in the early stages of the project.

Most random number generators are based on a simple numerical algorithm that will repeat a sequential pattern after some maximum number of variants is reached. The random number generator used as the base of the research done in this thesis had to be powerful enough to produce a large sample of variates without repeating the sequence. To achieve 99.73% degree of accuracy in the methods being used, a simulation on the order of one-hundred thousand assemblies must be performed. Therefore, if an assembly had twenty parts whose tolerances were being evaluated, the random number generator must produce at least two-million random variates without recursion. For higher degrees of accuracy this number increases to almost 40 million.

While the sample size is one important issue, another is the scattering of the representative distribution. The random number algorithm must produce truly random results to be useful for our purposes. If any recursive pattern arises due to the number generation scheme, the results will be biased due to this algorithm and not the tolerance distribution characteristics. This would lead to a poor approximation of the actual manufacturing process and a flawed analysis.

The algorithm that has been chosen for a uniform distribution is taken directly from Numerical Recipes [12,pg.195]. This algorithm is called **RANO** and is used to improve the Vax system supplied routine RAN. This is accomplished by performing a randomized shuffle. A copy of this algorithm, called **RANO**, is included in the Appendix. The result of the **RANO** algorithm varies uniformly about a midpoint, 0.5, and range from 0.0 to 1.0. To use this in our methods we do a simple mapping of the resultant variant to the specified tolerance range and add it to the nominal value. For example if a specified tolerance on a part were -0.002 inches to +0.002 inches and the nominal value was 3.0 inches then the tolerance range would be $\mathbf{Tol\ Rng} = 0.002 - (-0.002) = 0.004$. If the random variate were 0.75 then the resultant part length would be $\mathbf{Prt\ Lth} = 3.0 + 0.004(0.75 - 0.5) = 3.001$. The value of -0.5 is added to the random variate to produce a semi-tolerance as used in most tolerance analysis. This resultant fraction will be referred to later as Alpha.

A base random number algorithm is generally supplied by the computer system on which you are working. In this case the **RAN** function from the Vax

FORTRAN compiler was used as the base. The Vax function **RAN** was tested and after sixteen-million generated random variates the algorithm began to repeat. Therefore, the RAN function proved to be an inadequate method of feeding the modified algorithm. The basic **RAN** function was modified, with the **RANO** algorithm and tested. The modified uniform generated repeated after 10.5 million variates. This method of variate generation proved to be a sufficient base for the statistical techniques tested in this paper.

3.2 Non-Normal Distributions

To allow for greater flexibility of analysis for many manufacturing processes, the new methods discussed in this thesis must provide a method of generating non-Normal distributions. Many manufacturing processes have characteristic distributions that are non-Normal, such as casting, powder metallurgy or injection molding. Figure 3.1 demonstrates how a process can be skewed due to tool wear. Figure 3.1 also demonstrates the importance of correctly modeling a non-Normal process. If a simple Normal distribution were used to approximate this process the resulting analysis would be far from helpful.

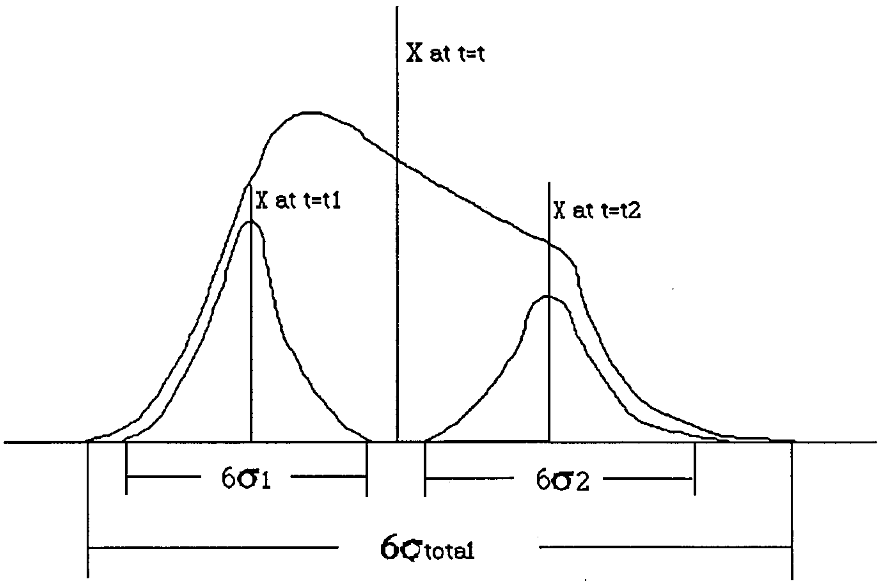


Figure 3.1 Non-Normal Distribution Due to Tool Wear

Most statistical distributions can be classified by their physical shapes. Two factors effect the shape significantly, the skewness and kurtosis. These are non-dimensional values representing the asymmetry and the peakedness of the distribution. The skewness and kurtosis, sometimes referred to as β_1 and β_2, will be discussed in greater detail later in this paper. There are several different families of statistical distributions with variable skewness and kurtosis. Some of these are: Weibull, Johnson, Lognormal, Beta, Gamma, and Generalized Lambda. All of these distributions have different limitations and characteristics. Hahn and Shapiro[9,pg. 221] present a valuable graphical comparison relating Skewness to Kurtosis. In figures 3.2 and 3.4 the shape ranges of various distribution families are compared. These charts demonstrate the various shape possibilities of each representative family. Figure 3.2 also presents some well-known distributions and their locations on the Skewness verses Kurtosis chart.

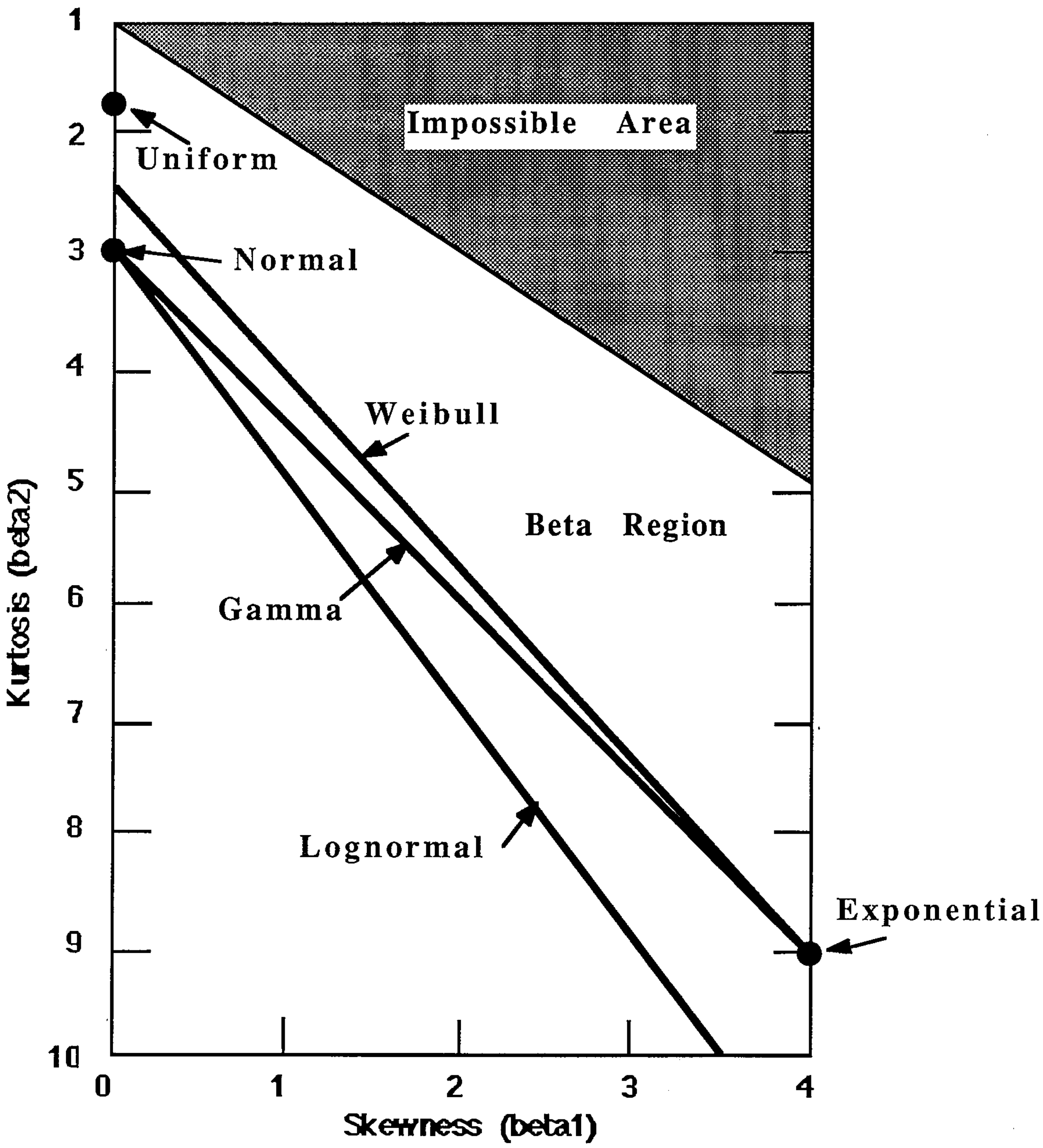


Figure 3.2 Skewness vs. Kurtosis Plane for Various Well Known Distributions

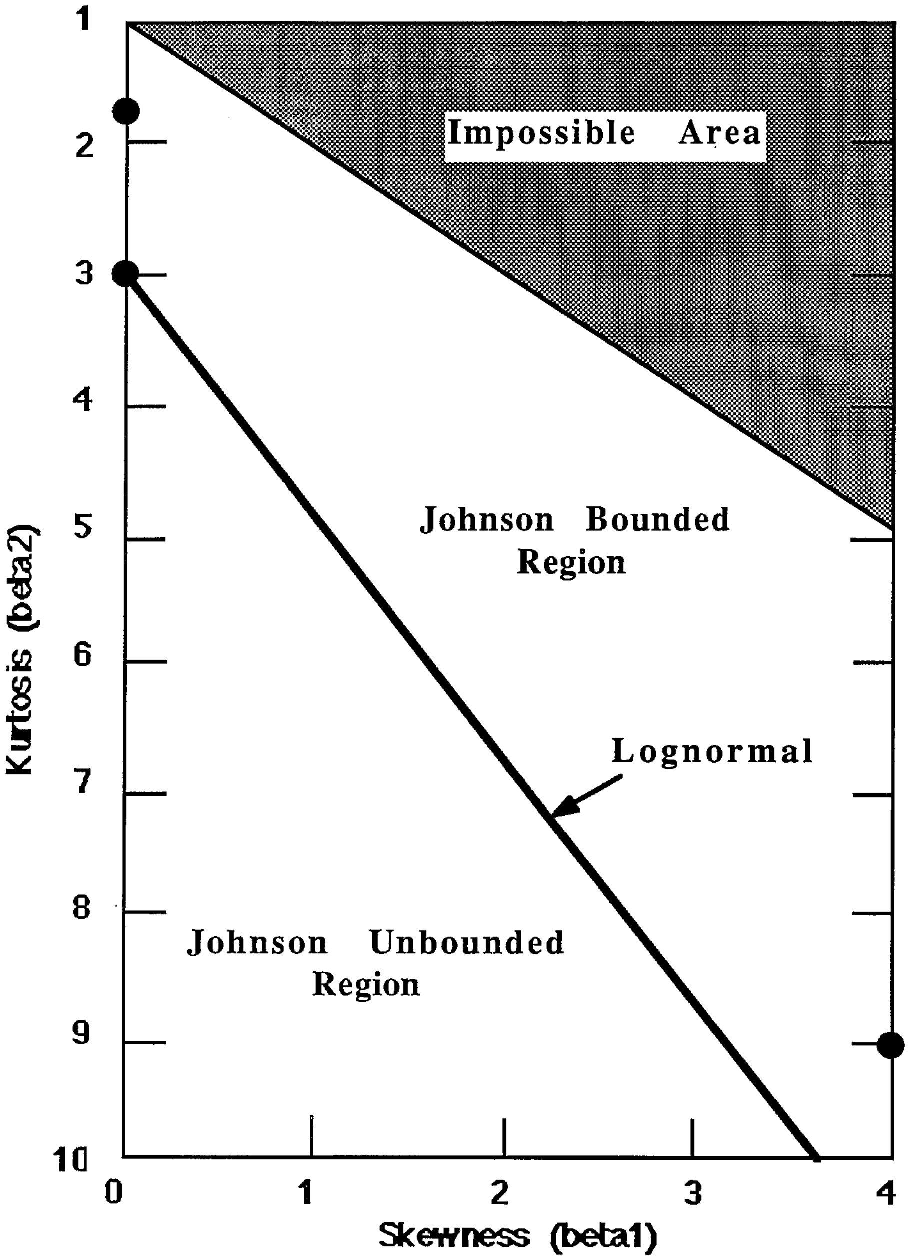


Figure 3.3 Skewness vs. Kurtosis Plane, Johnson Distribution

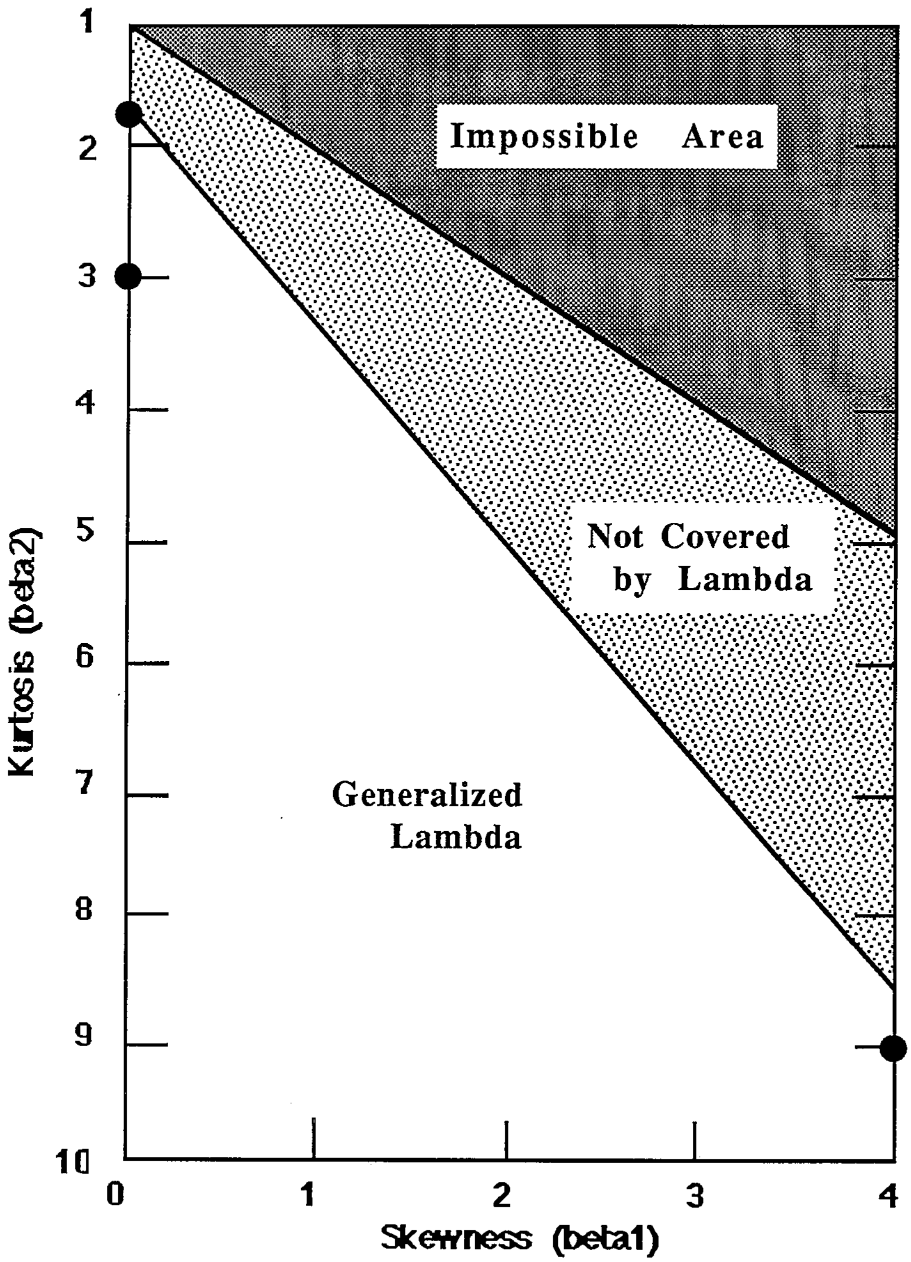


Figure 3.4 Skewness vs. Kurtosis Plane, Lambda Distributions

From the graphical comparisons the Johnson family appears to be the best choice, however, the algorithm to generate the Johnson family does not lend itself to an iterative computer use. This is due primarily to the application of three separate methods of generation for the three areas shown in figure 3.3. It also lacks an empirical representation. The Generalized Lambda appears to be the most applicable and straight forward. The Lambda empirical distribution covers a wide range of distributions with only one general form, Equation 3.1.

$$X(\mathbf{p}) = \left[Lam_1 + \frac{\mathbf{p}^{Lam_3} - (1.0 - \mathbf{p})^{Lam_4}}{Lam_2} \right] \qquad \text{Equation 3.1}$$

where: X(**p**) is the percentile function, $Lam_{1\text{-}4}$ are shape factors and **p** is the probability factor.

The Generalized Lambda Distribution (GLD) [14] is defined in terms of a percentile function (equation 3.1), where the percentile function of a distribution is defined as $X_p = F^{-1}(p)$ and the value X_p must satisfy equation 3.2 [16,pg. 171].

$$P = \int_{-\infty}^{X_p} f(u)\, du \qquad \text{Equation 3.2}$$

However, to use the GLD effectively, a method of generating various distributions based upon the density function, not the percentile function, must be developed. This would create a means for the characterization of manufactured parts based on their statistical distribution and the computerized generation of those distributions. Shapiro and Gross define a relationship between the percentile function and the cumulative density function [16, pg. 179] such that the cumulative density function is the inverse first derivative of the percentile function. This relationship is described by equation 3.3, where f(x) is the density function and R(p) is the percentile function.

$$f(x) = \frac{1}{R'(p)} \qquad \text{Equation 3.3}$$

The cumulative density function for the Generalized Lambda Distribution can be obtained through the application of equations 3.3 and equation 3.1.

$$f(x) = \frac{Lam_2}{Lam_3\, \mathbf{p}^{(Lam_3 - 1)} - Lam_4 (\mathbf{1} - \mathbf{p})^{(Lam_4 - 1)}} \qquad \text{Equation 3.4}$$

Equation 3.4 is especially suited to Monte Carlo simulation since Monte Carlo is based on random variation. If **p** is generated by a uniform random number generator, whose range is between zero and one, the simple mapping of **p** onto the density function will produce random numbers which describe a specific distribution (see figure 3.5). A wide variety of distribution shapes are possible. These shapes are characterized only by the factors Lam_1 through Lam_4 of equation 3.4, which in turn relate directly to the first four statistical moments. Therefore, by using equation 3.4 with a simple uniform random number generator, it is possible to simulate virtually any type of manufacturing process.

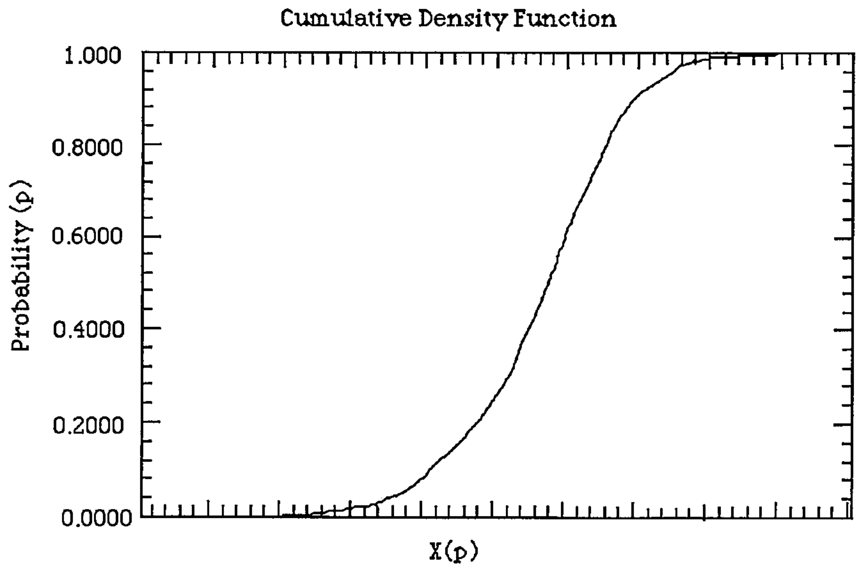


Figure 3.5 Cumulative Density Function.

The values of the Lambda parameters have been tabulated for a wide variety of distribution shapes [14]. An interpolation scheme is used as a general means to extract shape values from the stored table. As for further research, it may be possible to curvefit the tabulated Lambda parameters vs. distribution moments and use this information for fitting the general case.

To use the generalized Lambda distributions as an effective tool it was decided to have a set of common "Pre-Set" values representing common range of distributions. This would ease the strain placed on the designer to calculate the system moments for each component distribution from the real data It would also enhance the design process for simple systems since an approximation is most likely sufficient. For the special case where a part must be approximated with a specific distribution not defined by the "Pre-Sets", the designer must characterize the manufacturing processes by calculating the systems moments and non-dimensionalizing them for use with the generalized Lambda table. These Pre-Sets distributions are compared to Normal and displayed with their β_1 and β_2 values in figure 3.6.

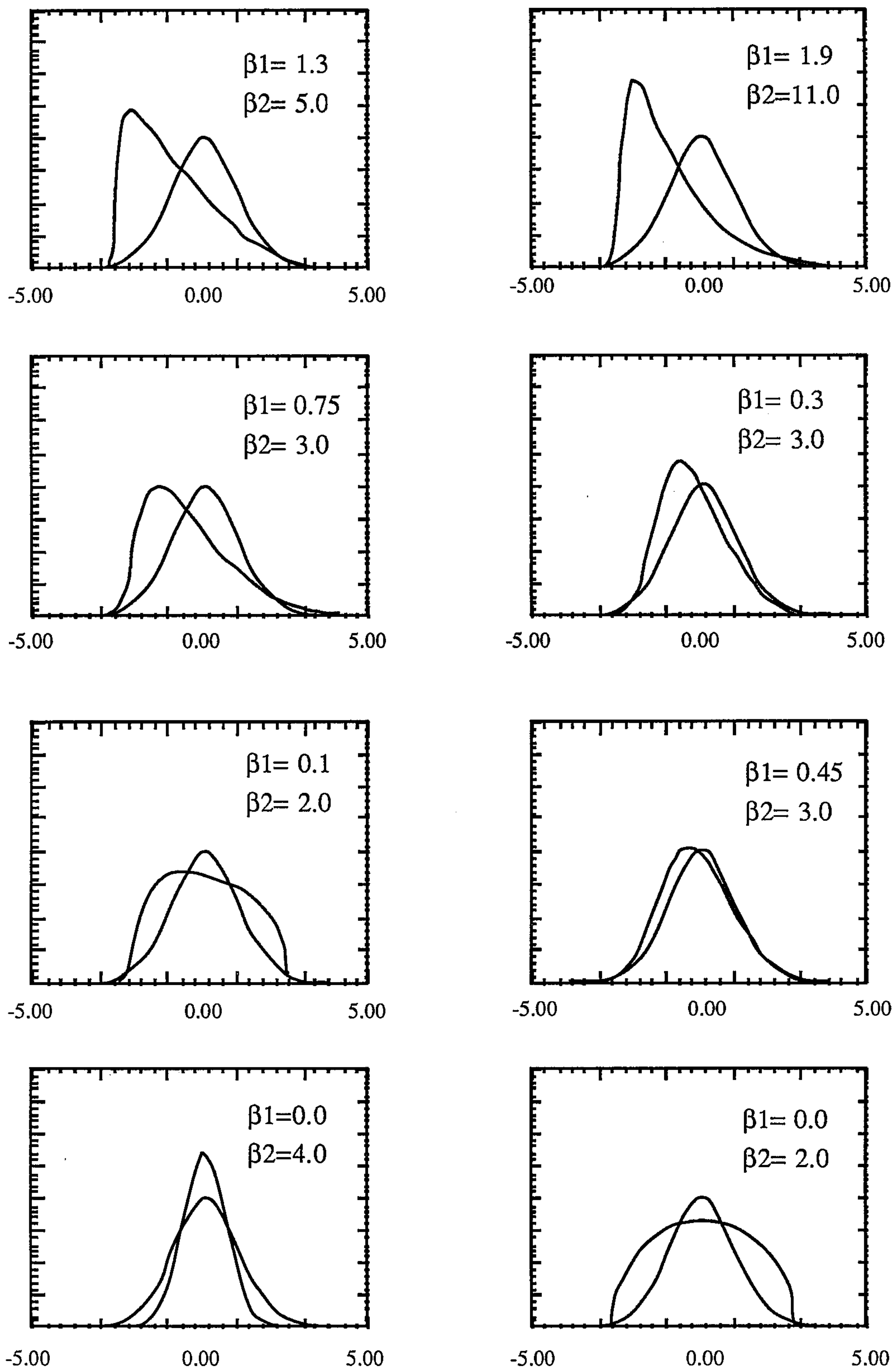


Figure 3.6 Pre-Set Distributions.

4. Basic Monte Carlo Simulation and Tolerance Analysis

4.1 An Historical Prospective

From late in the nineteenth century until the 2nd World War, the centralized, one-man control of a product from design to assembly gradually shifted to the complex production system by specialists that we have today. This decentralization brought with it the separation of design and manufacturing. It became more and more difficult for one person to control the output of such a system, resulting in added emphasis being placed on communication through the use of engineering drawings.

It was the intense production demands brought about by the war effort that forced the British to evaluate the wasted resources due to improper part assembly. The British concluded that much of the problem was due to lack of complete process information on the engineering drawings. This situation lead the British to re-examine drawing practices and develop the theory of true position tolerance analysis. The British continued to develop methods of tolerance analysis and in 1948 the first standards of tolerancing were published in the United Kingdom.

The first American industries to recognize the importance of tolerancing were the aircraft and defense companies. Because of the widespread use of these methods, the Generalized Dimensioning and Tolerancing (G.D.&T) standards were developed.

The use of Monte Carlo techniques with tolerance analysis by the U.S. Military was first implemented after World War 2. The simple simulation techniques developed after the war underwent extensive modification from 1966 through the 1970's at the Chevrolet Product Assurance Department of General Motors. These refined procedures, consisting of more versatile and precise modeling techniques as well as more accurate statistical methods, were renamed Variation Simulation Modeling (V.S.M.)[10].

This was the birth of advanced statistical tolerance analysis in the United States. Since that time many successful analyses have been performed through the use of statistical tolerance analysis. One of the earliest examples of General Motors use for tolerance analysis would be the front end sheet metal of the General Motors "S" series trucks. These procedures were so successful that shortly after their initiation on the "S" series trucks, a more aggressive project was planned for the suspension of the 1982 Corvette [10].

4.2 Tolerance Analysis

Elementary methods of tolerance analysis have been found to be inadequate in most cases, particularly where component distributions are non-Normal or have a mean shift. When used improperly elementary methods often result in higher production costs and rejection rates [8, pg 6].

For the more difficult tolerancing problems there are advanced methods of analysis. These include Method of System Moments, Monte Carlo Simulation, and Hasofer-Lind Reliability Index. The Hasofer-Lind method will not be discussed in this paper since it is limited to Normal distributions. In this section the basic concepts of Monte Carlo will be introduced and applied to tolerance analysis.

4.3 Basic Monte Carlo

Monte Carlo simulation is a device for studying an artificial model of a physical or mathematical process. Once a system or process is modeled mathematically, it can be evaluated for varying configuration parameter. When using Monte Carlo, a system is examined through repeated evaluation of the mathematical model as its parameters are randomly varied according to their expected behavior. The mathematical model may describe various systems either linear or non-linear. Modern day computers use random number generation to simulate a system by evaluating the mathematical model for numerous sets of random samples drawn from the parameter distributions. Measurements of resultant dimensions are accumulated and the resultant distribution is examined to evaluate the systems performance. If the performance is not satisfactory, the parameters are adjusted and the simulation is redone. This process is repeated until adequate results are attained [7].

Monte Carlo based tolerance analysis overcomes the shortcomings of both the worst-case and RSS techniques. Often it is the best way to cope with assemblies of parts whose variation may not fit a Normal distribution, or with parts whose variation may not have linear effects on the performance of the system [3]. With this method component tolerance distributions are represented by the generation of random variates. The mathematical model is a representation of the physical assembly process used in the construction of a part. Through the assembly function, variations in the assembly dimensions can be determined by the variation of the component dimensions. Repeating this simulation many times approximates the real manufacturing process.(See figure 4.1).

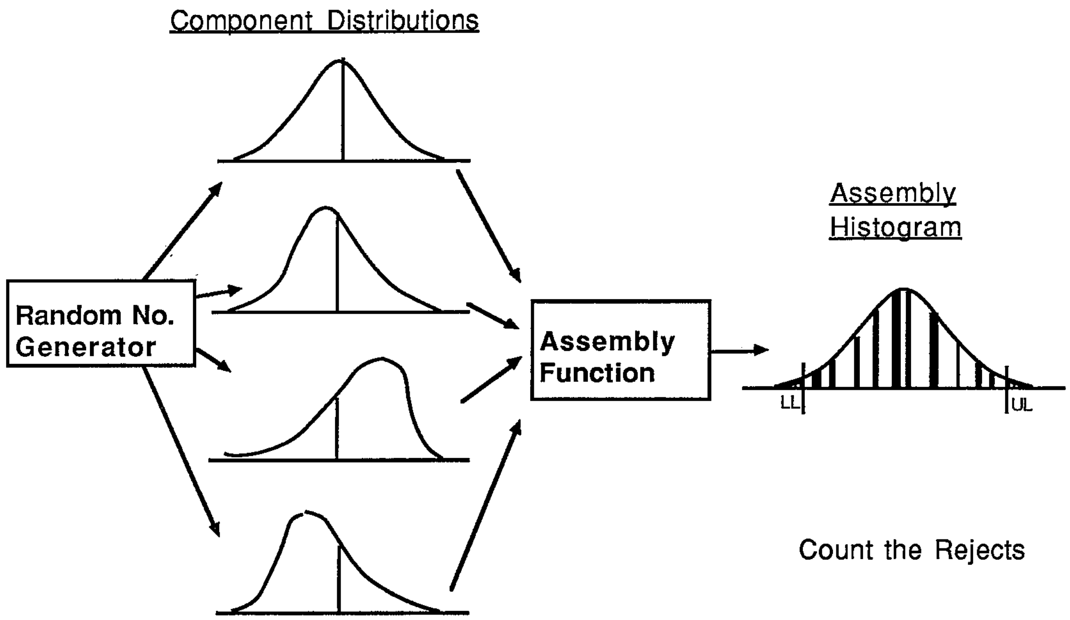


Figure 4.1 Basic Monte Carlo Simulation

After each assembly simulation the dimension of interest is compared to the predefined tolerance limits. Those dimensions falling outside of the limits are counted as rejects. The reject fraction is then calculated by dividing the total number of rejects by the number of simulations.

To insure accurate results using Monte Carlo techniques it is generally necessary to perform a large number of simulations. This is primarily because of the two following reasons. One, since the component dimensions are generated using a random number algorithm, the rejection fraction will vary with the number of assembly simulations. Two, for an efficient design, the rejection fraction is usually quite small [8, pg 38].

The step-by-step method of using Monte Carlo for tolerance analysis is:

1) Classification of the manufacturing process associated with an the assembly components according to a specified distribution type.

2) Identify critical assembly dimensions and define the assembly function.

3) Generation of the random variates for the component. These variates are scaled to the desired tolerance range and added to the nominal component dimension.

3) Random configuration of the components to produce a completed assembly. Using the assembly function, the random component dimensions are combined, simulating the actual assembly process.

4) Comparison of the assembly to a specified acceptable tolerance range and calculation of the reject fraction.

Repeating this process many times will simulate the actual manufacturing process. It is important to realize that many processes can not be accurately represented by a Normal distribution. For this reason, both Normal and non-Normal variate generation must be developed. This will provide greater generality and accuracy in the simulation.

It is also important to emphasize that Monte Carlo simulation is only as good as the random number generator driving it. Therefore, it is crucial to verify the random number generation to ensure accurate results. After it has been determined that the number generation is satisfactory, verification of the Monte Carlo package has been done by:

1) Comparing simulated Normal distribution results to classical RSS statistical results.

2) Comparing simulated non-Normal distribution results to known statistical results.

3) Comparing the Lambda equivalent of Normal distribution results to the statistical results.

4) Using various text book problems.

4.4 Advantages of Monte Carlo

The Monte Carlo model, when integrated with statistical theory, is effective in predicting assembly problems resulting from variation inherent in all manufacturing processes. Typically, the simulation model predicts variation in the resultant assembly through the assignment of tolerances to critical part dimensions. These tolerances consist of process variations that are introduced by manufacturing and assembly operations and their interaction with the tolerances of the individual parts in the assembly.

In many situations it is impossible to correctly approximate the manufacturing process with a simple Normal distribution. It is likewise difficult to mathematically describe all assembly situations linearly. In these cases Monte Carlo provides an excellent method of modeling and analyzing the more difficult non-Normal distributions as well as the non-Linear assembly functions.

4.5 Disadvantages of Monte Carlo

While Monte Carlo is conceptually simple and allows for difficult distributions and non-Linear assembly functions, it is not without some draw-backs. The worst of which is the computer intense algorithms that must be used in order to achieve the desired accuracy. To attain the accuracy needed for this type of analysis, simulations on the order of one-hundred thousand and higher are required. The expression for the required sample size is presented in equation 4.1, based on a statistical analysis [16].

$$N = \left[\frac{\theta'(1.0 - \theta')}{D^2} Z^2_{1-\alpha/2}\right] + 1.0 \qquad \text{Equation 4.1}$$

Where :

θ' = initial estimate for θ
D = half band width of the error band
$z_{1-\alpha/2}$ = normal deviate corresponding to a confidence coefficient of $1-\alpha$
N = sample size.

A table of the required sample size for acceptance and accuracy of estimate rejects is shown in Table 4.1.

Table 4.1 Sample Size Requirements for Monte Carlo.

Assembly Yield	Error in Rejects (±) 5%	10%	25%
.99	107,000	27,000	4,300
.995	217,000	54,000	8,600
.9973	400,000	100,000	16,000
.999	1,081,000	270,000	43,000
.99999	10,823,000	2,706,000	433,000
Sample sizes are for 90% confidence interval			

Multiplying the number of simulations by the number of parts per assembly, it is clear to see that Monte Carlo does not lend itself to design iteration. A slight change in the part information or assembly function would constitute a large amount of computer time due to the fact that the whole simulation must be repeated to analyze the change in design. The information in Table 4.1 was generated using equation 4.1 [16, pg 292] . For example, taking a process of ten-thousand assemblies with a designated yield of 0.9973, twenty-seven rejects would be expected. Using Monte Carlo to simulate the, a calculated yield of 0.9968 was obtained. This would represent thirty-two rejected assemblies, a 18.51% error in estimated rejects. Therefore, a difference between the actual and calculated acceptance fraction in the fourth decimal place results in a large error in the number of projected rejects.

Another drawback that might seem less obvious, but is still a major difficulty, is the requirement placed on the designer to have a precise knowledge of the manufacturing process. This knowledge is needed in order to characterize each component accurately, thus enabling an appropriate approximation of the overall assembly procedure. Without the detailed information of the part distribution it is impossible to use Monte Carlo techniques effectively. Unfortunately, in the design stage of a project many designers do not have access to the precise manufacturing information needed to make a valid assumption about distribution characteristics of a part. This greatly limits the use of Monte Carlo as a design tool.

5. Hybrid Monte Carlo Simulation

5.1 Statistical Characteristics

The characteristic shape of a distribution may be represented by its moments. This assumption will not cause any major difficulties since it has been shown that any arbitrary distribution may be uniquely defined in terms of its moments [8].

The moments of a statistical distribution can be found by integrating the density function and multiplying it by the difference between the mean and the given value according to equation 5.1. Where **M_K** is the **K**th moment about the origin and **X** is some random value (see figure 5.1).

$$M_k = \int_{-\infty}^{\infty} (X - \overline{X})^k \, f(x)\, dx$$

Equation 5.1

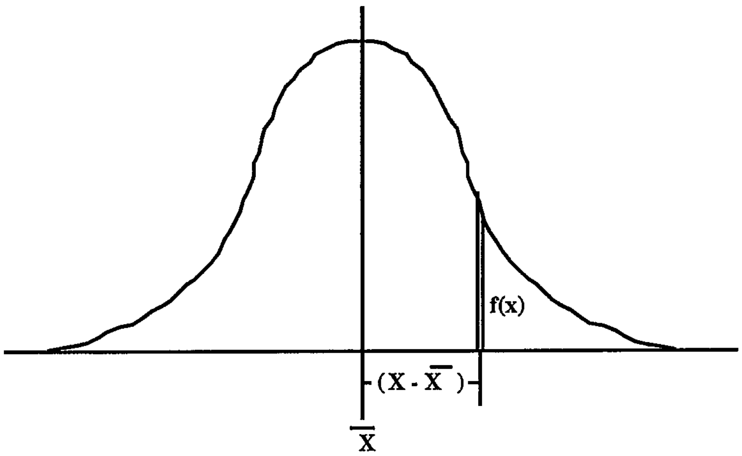


Figure 5.1 Determination of System Moments

If the data X is sampled discretely, it is possible to numerically integrate the moment equation 5.1 by using a simple summation as shown in equation 5.2. Where N is the number of samples taken, $\mathbf{X_i}$ is the present sample value and $\overline{\mathbf{X}}$ is the mean value of the population.

$$M_k = \frac{1}{N}\sum_{i=1}^{N} \left(X_i - \overline{X}\right)^k \qquad \text{Equation 5.2}$$

As was mentioned earlier, for most cases only the first four moments are required to sufficiently approximate the distribution [8]. A sample of the first four moments and their contribution to the shape of the density function is shown in figure 5.2. The first moment about the origin is referred to as the mean. This is a measure of the distribution's nominal value or central tendencies. The second moment is the variance, which is a description of the distributions scale or spread. The third moment is the skewness and it gives a measure of symmetry. The fourth moment is called the kurtosis and it defines the peakedness of the distribution. There are other higher moments that help in the exact description of a distribution, but most cases it is sufficient to use only the first four moments to characterize a distribution accurately.

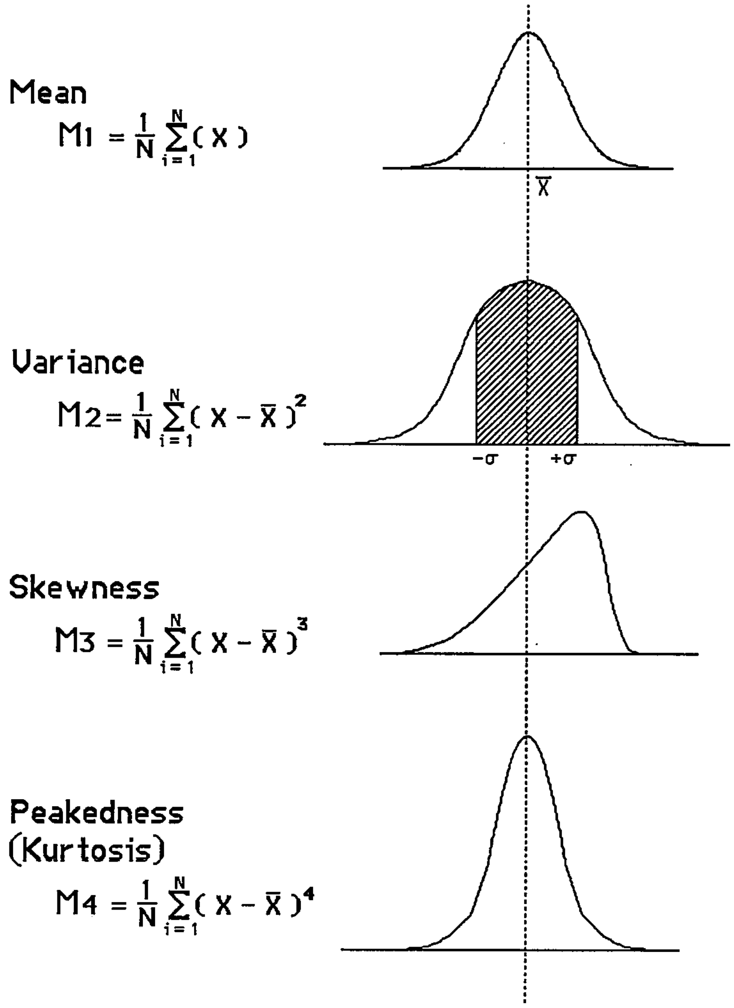


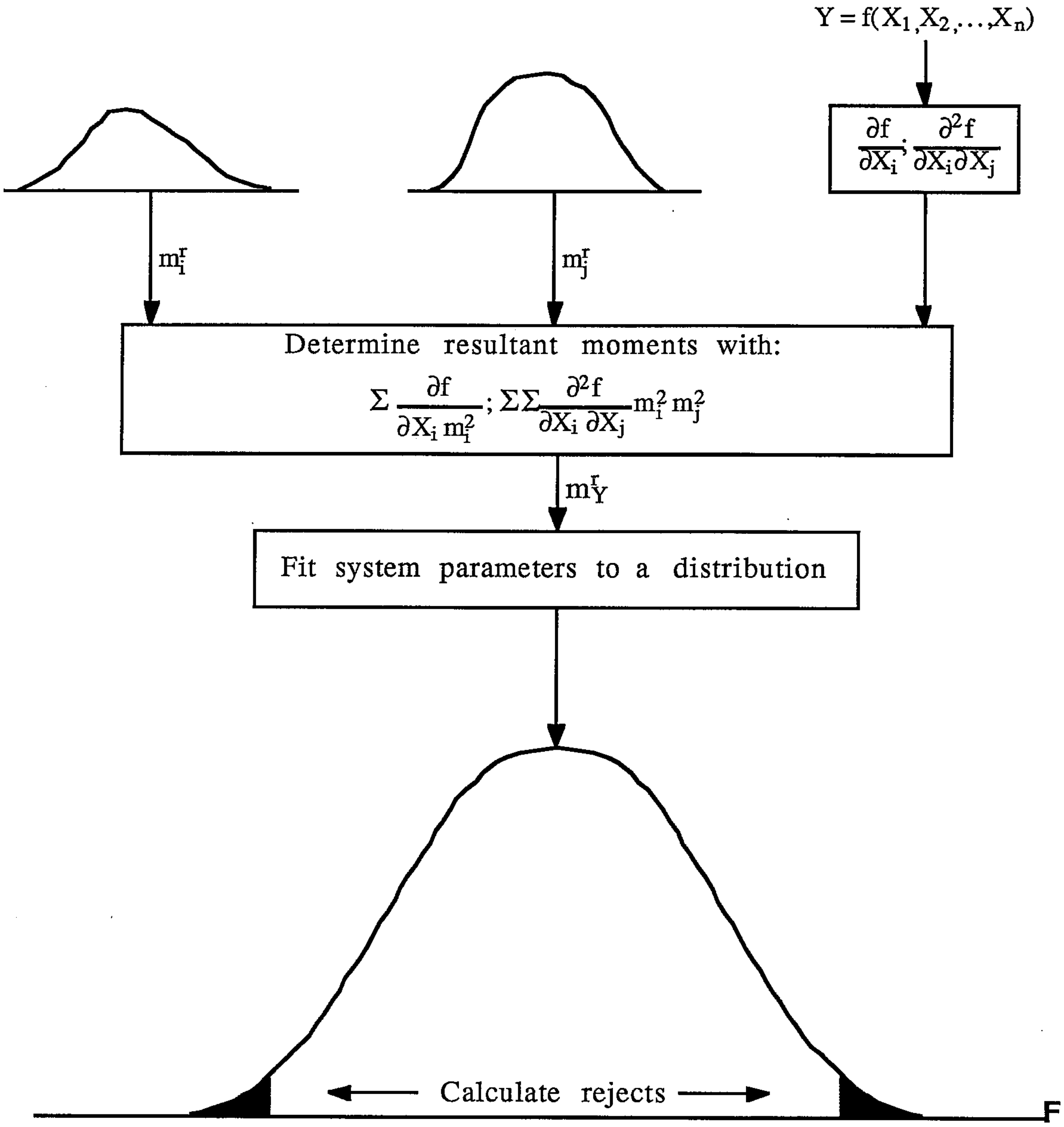
Figure 5.2 Statistical Moments

5.2 Method Of System Moments

The Method of System Moments uses the statistical moments of the individual component distributions to determine the characteristic distribution of an assembly of those component parts. This method uses a second order Taylor's series approximation of the assembly function to determine the first four moments of the desired assembly distribution [16,pg. 296-304]. These expressions are complex and extensive and will not be presented here [9,pg. 42]. Greenwood [8,pg. 21] points out that for non-linear assembly equations this approximation requires the first partial derivatives and the second partials, the Hessian. Since this method requires the use of second partial derivatives, an approximation for the first eight moments of the components are needed in order to accurately calculate the first four moments of the assembly. In the special case when the assembly function is linear, the higher moments may be eliminated since all of the second derivative terms are zero. In this special case the approximation for the system moments becomes exact and the calculation of the systems moments becomes greatly simplified.

For tolerance analysis applications, the resultant assembly distribution is generally simulated with a Normal distribution and the assembly function is represented by a linearized sum using the first partial derivatives. In this specific case only the first four moments of the component distributions are needed to analyze the process.

After the moments have been determined, the distribution of the resultant assembly can be approximated with a known distribution. It is then a simple task to calculate the percentage of assemblies that fall outside of the specified acceptance range by simple integration or the use of tables. This process is represented graphically in figure 5.3.



igure 5.3 The Method of System Moments

5.3 Advantages of the Method of System Moments

The computational time used for the Method of System Moments is significantly less than that used in Monte Carlo simulation since the distributions are not simulated using random variates. The evaluation of derivatives and series expansions are often only hundreds of operations per independent variable versus tens of thousands needed for Monte Carlo. This method also predicts accurate results for all but highly non-linear assembly functions[8, pg. 85].

5.4 Disadvantages of the Method of System Moments

Like most of the statistical techniques, the Method of System Moments requires a fairly detailed knowledge of each component's manufacturing process. This knowledge is essential in the correct modeling of each component. Generally, this type of detailed data is not known before production begins. Also, since the Method of System Moments is based on the use of second derivatives in calculating the system moments, problems with non-linear assembly functions become significantly more difficult.

5.5 A Hybrid Method

The Hybrid method incorporates the positive aspect of both Monte Carlo and the Method of System Moments. It has been proposed that when using this method a more accurate and efficient simulation method is generated.

By combining Monte Carlo with the Method of System Moments, accurate results for almost any type of system can be attained. The difficulties encountered with non-linear assembly functions and non-Normal distributions when using the Method of System Moments is eliminated by the Hybrid technique. In the Hybrid method the first four moments of the assembly distribution are calculated using the component dimensions created by Monte Carlo and equation 5.2. These moments are then used to fit the assembly function with a known statistical distribution, just as was done in the Method of System Moments. It is then possible to estimate the percent rejects by numerical integration or from tabulated data. Since the Hybrid method only uses Monte Carlo to determine the shape of the resultant distribution, accurate results can be obtained for all types of assembly functions and distribution types. Thus, it is expected that non-linear assembly functions and non-Normal distributions can be accurately modeled with less computer time than would be needed using traditional Monte Carlo simulation.

A flow chart of the hybrid method is shown in Figure 5.4.

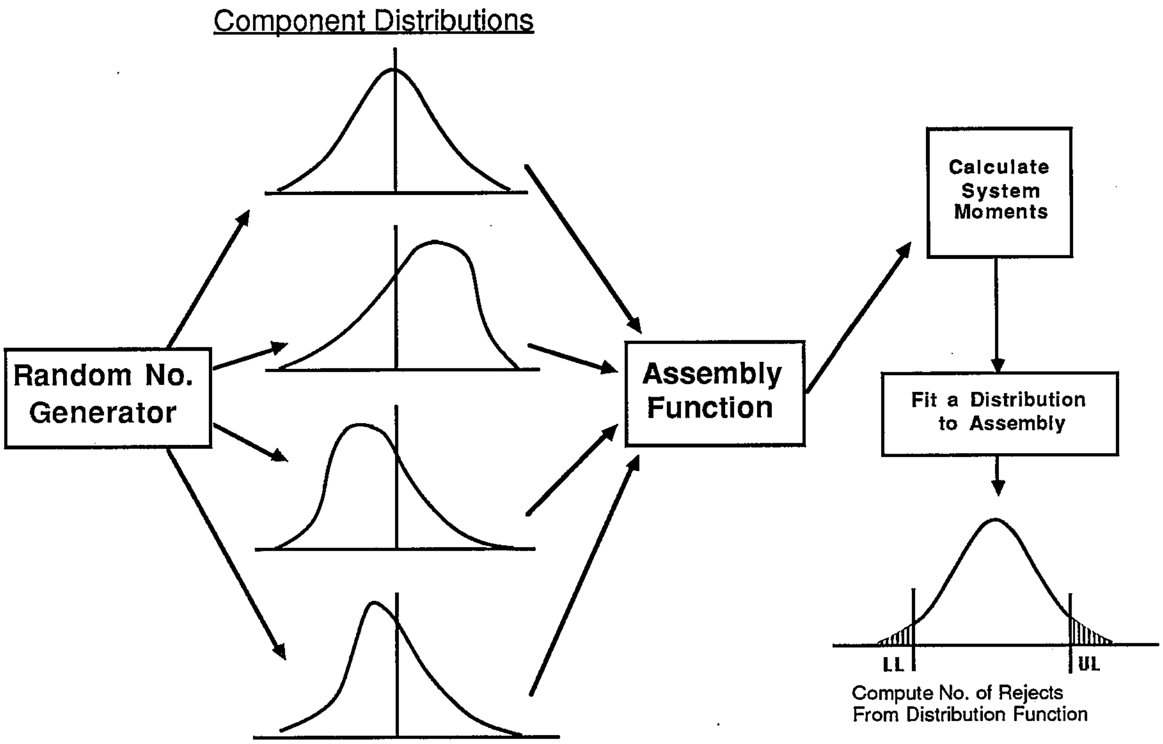


Figure 5.4 Hybrid Method

It is also important to determined the effectiveness of the Hybrid methods in the design environment. Effectiveness was checked by comparing efficiency and accuracy as a function of sample size and by comparing the system output to the traditional methods. The results of this analysis is discussed in Chapter 7.

5.6 Advantage of the Hybrid Method

Unlike Monte Carlo the Hybrid method allows the calculation of a functional description of the assembly distribution rather than an empirical one. With this functional relationship the rejection percentage can be calculated exactly rather than counting each reject as in Monte Carlo. In addition, if the assembly limits are altered, the new rejection percentage can be calculated without repeating the simulation.

5.7 Disadvantage of the Hybrid Method

The major disadvantage of the Hybrid method is that it does not allow the modification of component distribution information without repeating the entire simulation. If assembly limits are prescribed by performance requirements, the only way to obtain a specified yield is to alter the component distribution information iteratively until the resultant assembly distribution meets the yield requirements. Thus, the Monte Carlo simulation must be repeated iteratively.

6. Optimal Design Simulation Technique

Thus far in this paper various methods of tolerance analysis and the uses of statistics have been discused. The subject of design, however, has not been covered. There are a few subtle but important differences between analyses and design. Analysis is based on the assumption that all input parameters are specified and that the system under investigation can be accurately modeled mathematically. With the input parameters and the model, the output of the system is calculated. The output of the system can then be evaluated in terms of the specified inputs. Based on this evaluation it is possible to determine the acceptability of the design (see figure 6.1).

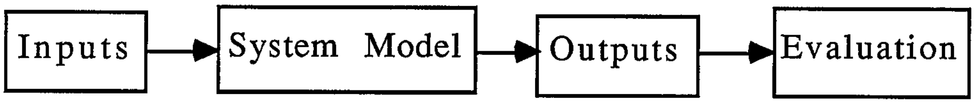


Figure 6.1 Flow Diagram of the Analysis Process.

Design is also based on the assumption that systems can be modeled mathematically. However, in the design domain the inputs are handled as variables and the output is pre-specified. The output of the system is calculated based on the variable inputs and compared to a pre-defined output value. If this comparison falls outside of a specified range the input variables are adjusted. This process is repeated until the calculated output is acceptable. The final values of the inputs are said to be the design parameters (see figure 6.2).

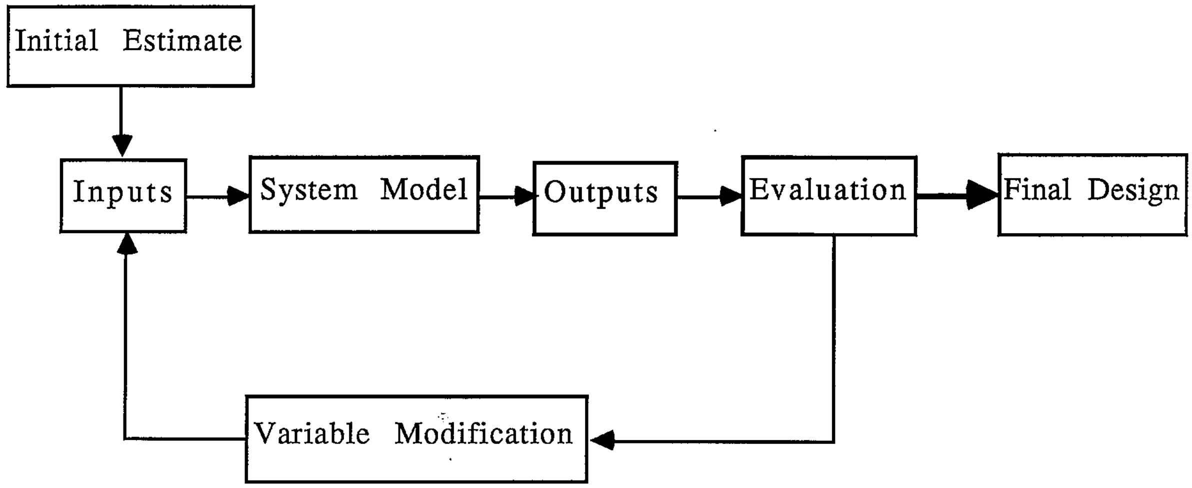


Figure 6.2 Flow Diagram of the Design Process.

In tolerance design the designer generally specifies a desired acceptance fraction for the assembly distribution. He then selects some initial tolerances for the components that he knows to be close to an acceptable value. The simulation is performed and based on the results of that simulation the initial design is evaluated. The input tolerances are re-allocated and the simulation is repeated until the design meets the specified yield requirements.

6.1 Optimal Design

A new and more effective method of tolerance design called the Optimal Design Simulation Technique, has been developed by modifying the Hybrid Method and incorporating a family of distributions. Even though the Hybrid Method can model most difficult problems accurately, if it is desired to use this method for design purposes, one main obstacle still exists. The Hybrid method is too computer intensive for the iterative design environment. To achieve an accurate simulation of one-hundred-thousand or more assemblies, the computer time would become very costly. This along with the conceptual difficulties is what deters most designers from using the hybrid method as a design tool. The proposed design method eliminates the cost involved with design iteration by storing the random variates in a compact non-dimensional form, eliminating the need to create a new set of variates after each iteration. When a design is modified the method will use the stored distribution shape information and re-simulate without once again calculating each variate. By saving the results of the initial simulation and using the tolerances as design variables, the accurate Hybrid Method can be used as an effective design tool.

The first step in Optimal Design Simulation is to select the initial tolerances and distribution shapes for the components of the assembly. With the use of a random number generator and the Generalized Lambda equation, most useful types of distributions can be approximated to produce the discrete data needed for Monte Carlo simulation. Using Monte Carlo, the system moments of the assembly distribution are calculated based on specified component tolerances and the important shape information is stored in non-dimensional form. The Lambda parameters are then interpolated from a table of stored values by non-dimensionalizing the first four statistical moments of the assembly distribution. After retrieving the non-dimensionalized Lambda values they are re-dimensionalized and used in equation 3.2 to approximate the assembly distribution. The percent rejects are then calculated using equation 3.3. If the rejection fraction is not acceptable the tolerances are reassigned and the process is repeated until an acceptable design is achieved. The Optimal Design Simulation process is shown in figure 6.3.

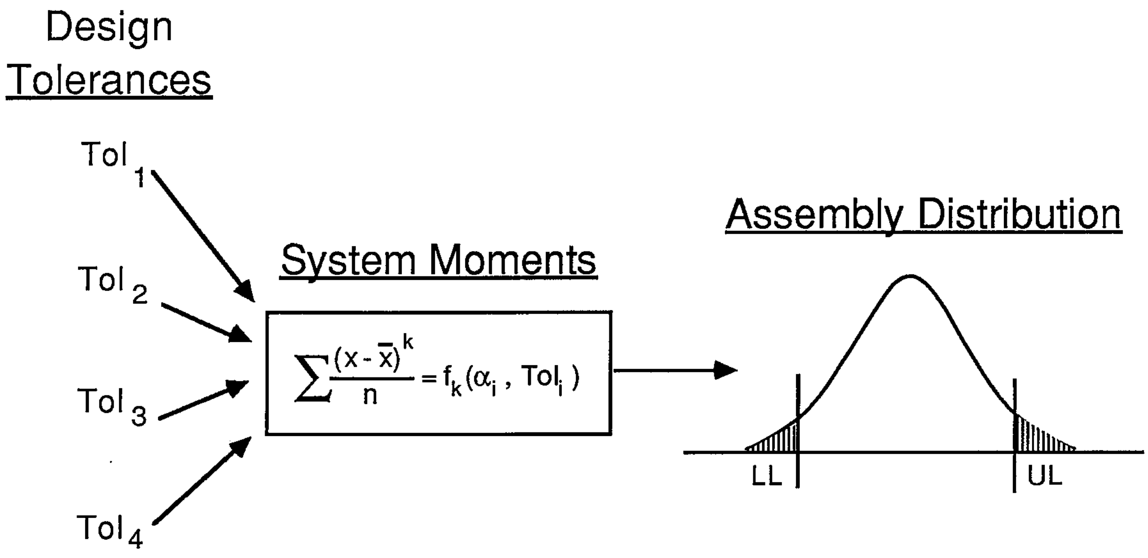


Figure 6.3 Optimal Design Simulation

6.2 Uncoupling Tolerances from the System

The main problem encountered when using Optimal Design Simulation as a design tool is the uncoupling of the tolerances from the random variates and using those tolerances as design variables. To accomplish this, the moments of the component distribution must be expressed as a polynomial function of the tolerances. By non-dimensionalizing each component variate and expressing it in terms of a percentage of the component tolerance it is possible to uncouple the tolerance from the component dimension. This is described by the fallowing derivation. Start with the basic moment equation 6.1, letting Y represent the assembly and X represent the components of that assembly.

$$M_K = \frac{1}{N} \sum_{i=1}^{N} \left(Y_i - \overline{Y}\right)^K \qquad \text{Equation 6.1}$$

Using the Taylor series expansion of equation 6.2, the expression of the moment for the assembly can be reduced to an equation based on the variation of the components of that assembly.

$$Y - \overline{Y} = \sum_{i=1}^{N} \left(\frac{\partial f}{\partial X_i}\right)(X_i - \overline{X}) + \sum_{i=1}^{N} \frac{1}{2}\left(\frac{\partial^2 f}{\partial X^2{}_i}\right)(X_i - \overline{X})^2 + \sum_{i=1}^{N-1} \sum_{j>i}^{N} \left(\frac{\partial^2 f}{\partial X_i \partial X_j}\right)(X_i - \overline{X})(X_j - \overline{X}) + \cdots$$

Equation 6.2

Neglecting the higher order terms of equation 6.2 and substituting it into equation 6.1 results in the simplified moment equation 6.3.

$$M_K = \frac{1}{N} \sum_{i=1}^{N} \left(\sum_{j=1}^{M} \left(\frac{\partial f}{\partial X_j}\right)(X_j - \overline{X}) \right)^K$$

Equation 6.3

Next, the tolerances are uncoupled from the system by calculating the percentage of the total tolerance that is represented by the component's dimension. In figure 6.3 it is graphically demonstrated how the tolerances, when multiplied by a random scaling factor from -1.0 to 1.0, can be related directly to the component's dimension (X_p) and the statistical distribution.

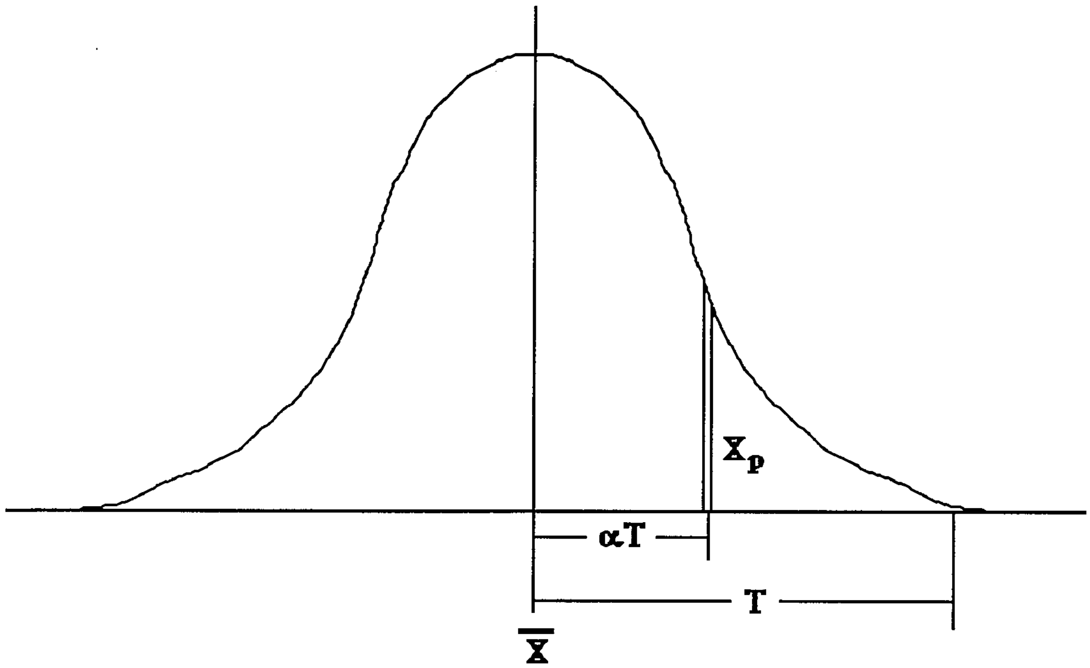


Figure 6.4 Tolerance Normalized Design.

Equation 6.4 is defined by substituting the relationship $X - \bar{X} = \alpha T$ into equation 6.3.

$$M_K = \frac{1}{N} \sum_{i=1}^{N} \left(\sum_{j=1}^{M} \left(\frac{\partial f}{\partial X_j} \right) \alpha_j T_j \right)^K \qquad \text{Equation 6.4}$$

Expanding the inner summation gives a polynomial expression in terms of the component tolerances T_j, the coefficients of which are the summations of the α_j for the simulation.

6.3 Example: Second Moment of a Three Part Assembly

For example, to find the second moment for an assembly of three parts, it is necessary to first substitute the summation of the component dimensions for the ith assembly in the simulation (equation 6.4). The result of this substitution is equation 6.5.

$$\text{StDev}=\frac{1}{N}\sum_{i=1}^{N}\left(\sum_{j=1}^{3}\left(\frac{\partial f}{\partial X_j}\right)(\alpha_j T_j)\right)^2$$

Equation 6.5

If we assume a linear assembly function, the value for all $\left(\frac{\partial f}{\partial X_j}\right)$ is one.

$$\text{StDev}=\frac{1}{N}\sum_{i=1}^{N}\left(\sum_{j=1}^{3}\alpha_j T_j\right)^2$$

Equation 6.6

Expanding the inner summation for the three components:

$$\text{StDev}=\frac{1}{N}\sum_{i=1}^{N}(\alpha_1 T_1+\alpha_2 T_2+\alpha_3 T_3)^2$$

Equation 6.6a

Expanding equation 6.6a and simplifying, the final polynomial expression for the second moment is shown in equation 6.7.

$$\text{StDev}=\frac{1}{N}\left[\begin{array}{c}\left(\sum_{i=1}^{N}\alpha_1^2\right)T_1^2+\left(\sum_{i=1}^{N}\alpha_2^2\right)T_2^2+\left(\sum_{i=1}^{N}\alpha_3^2\right)T_3^2+\\ 2\left(\sum_{i=1}^{N}\alpha_1\alpha_2\right)T_1T_2+2\left(\sum_{i=1}^{N}\alpha_1\alpha_3\right)T_1T_3+2\left(\sum_{i=1}^{N}\alpha_2\alpha_3\right)T_2T_3\end{array}\right]$$

Equation 6.7

Equation 6.7 results in a polynomial expression, where the coefficients contain all of the essential information about the standard deviation. In this case the tolerances may then be used as variable scale factors or design variables.

All of the Alpha coefficients are summed in a non-dimensional compact form. In this example, if 10,000 sample assemblies were simulated, the resulting six coefficients are all that must be stored to re-compute the second moment. By modifying one or more of the tolerances, the second moment of the new resultant distribution is easily re-computed without repeating the timeconsuming task of generating 30,000 new Alpha values. The acceptance fraction within the new distribution is then compared to the specified acceptance fraction. The process may be repeated until a satisfactory solution is found.

Initially the ODS method was designed to allow for forty parts with variable tolerances. The number of coefficients (C_k) that must be retained for the first four moments are expressed by equations 6.8a-c in terms of the number of components (M) in the assembly.

$$C_2 = \frac{M(M+1)}{2!} \qquad \text{Equation 6.8a}$$

$$C_3 = \frac{M(M+1)(M+2)}{3!} \qquad \text{Equation 6.8b}$$

$$C_4 = \frac{M(M+1)(M+2)(M+3)}{4!} \qquad \text{Equation 6.8c}$$

Table 6.1 shows the number of coefficients needed for each moment calculation for a twenty part assembly and a forty part assembly.

Table 6.1 Required Coefficient Storage

	Coefficients Required	
Moment	20 Parts	40 Parts
2nd	210	820
3rd	1,540	11,480
4th	8,855	123,410
Total	10,605	136,000

In the prototype computer program, the ODS method has been implemented. The capability of handling forty part assemblies was desirable, however, due to the large number of coefficients required for high part-count assemblies, especially in the higher moments, the number of allowable free choice tolerances was reduced to twenty. The ODS computer program did not reduce the total number of parts in an assembly only the number of parts with variable tolerances. It now allows twenty free and twenty fixed tolerances.

6.3 Non-Dimensionalized Alpha with Design

In Chapter three the problem of non-Normal distributions was discussed. In that chapter it was concluded that the Generalized Lambda family of Distributions (GLD) would lend itself quite adequately to Monte Carlo simulation. In case for Optimal Design Simulation, it can be incorporated with some slight modifications. The GLD family is based on four shape coefficients, $Lam_{1\text{-}4}$, and equation 3.2. For the general case these Lambda parameters must be non-dimensionalized. It is also necessary to derive the value for alpha in terms of the Lambda equation 3.2 and the shape parameters $Lam_{1\text{-}4}$. A component dimension, X_p, has been previously defined in terms of its nominal value $\overline{X}$, its tolerance range T, and some percentage of the tolerance range α.

$$X_p = \overline{X} + \alpha T \qquad \text{Equation 6.9}$$

From Chapter 3, X_p has also been defined in terms of the Lambda equation.

$$X_p = \lambda_1 + \frac{p^{\lambda_3} - (1.0 - p)^{\lambda_4}}{\lambda_2} \qquad \text{Equation 6.10}$$

Combining equations 6.9 and 6.10 gives a relationship between α and the dimensionalized Lambda parameters.

$$\alpha = \frac{1}{T}\left[\lambda_1 + \frac{p^{\lambda_3} - (1.0 - p)^{\lambda_4}}{\lambda_2} - \overline{X}\right]$$

Equation 6.11

For the general case, the Lambda parameters most be non-dimensionalized. This is done by scaling the specific system to a general system. For Normal distributions the variance is used for this purpose and the same is true for the generation of the GLD family of distributions. The data in the Lambda table supplied by Technometrics [14] are non-dimensionalized according to the variance. Letting λ^* represent the non-dimensionalized λ, the relationship for all the Lambdas can be defined as:

$$\lambda_1 = \lambda^*_1 \sqrt{\sigma^2} + \overline{X}$$ Equation 6.12a

$$\lambda_2 = \frac{\lambda^*_2}{\sigma}$$ Equation 6.12b

$$\lambda_3 = \lambda^*_3$$ Equation 6.12c

$$\lambda_4 = \lambda^*_4$$ Equation 6.12d

These definitions allow the translation of equation 6.11 to a non-dimensionalized relationship for α in terms of the Lambda parameters.

$$\alpha = \frac{\sigma}{T}\left[\lambda^*_1 + \frac{p^{\lambda^*_3} - (1.0 - p)^{\lambda^*_4}}{\lambda^*_2}\right]$$

Equation 6.13

This derivation can be taken one step further since the tolerance range, T, is most often specified in terms of the number of standard deviations or T=Zσ. Equation 6.13 reduces to the final equation 6.14 that describes the Alphas in terms of the non-dimensionalized Lambda parameters.

$$\alpha = \frac{1}{Z}\left[\lambda^*_1 + \frac{p\lambda^*_3 - (1.0 - p)\lambda^*_4}{\lambda^*_2}\right]$$ Equation 6.14

This relationship and the uncoupled moment expressions are the key to rapid retrieval and simulation of the Optimal Design technique.

7. Results

The main purpose of this research is to accurately model manufacturing variations, and predict their effects on assemblies of mechanical parts. This prediction may then be used to modify the input parameters of the model and improve the assembly process. Before the concepts developed within this thesis can be used as effective tools it is critical that the techniques be tested and verified. Verification of these methods will be performed by comparing the various methods and their outputs. For most of the verification, traditional Monte Carlo will be used as a benchmark. It is also logical to compare results to some classical examples such as an assembly of Normally distributed components.

Before the model verification is performed it is necessary to confirm some of the model parameters that relate directly to the accuracy of the simulation, but have no relation to the manufacturing process. Two of the parameters causing most concern in the system analysis are the sample size used to simulate the assembly process, and the number of components in each assembly.

7.1 Sample Size

As pointed out in Chapter 4, sample size is an important simulation model factor. The engineer performing the analysis must choose the best sample size on the basis of accuracy and cost. It is obviously desirable to have an accurate model which requires large samples. It is also important to have small sample sizes since large sample sizes correlate directly to time and expense. In this section the Optimum Design Simulation Technique (ODS) is compared to traditional Monte Carlo simulation for various sample sizes. The results of this section are displayed in tables and by graphical comparison.

The system being analyzed is an assembly consisting of one component. An assembly of known inputs and outputs can be established with the use of a single component distribution. The component distribution is highly non-Normal with a value of -1.9 for $\beta 1$ (Skewness) and 11.0 for β_2 (Kurtosis). Table 7.1 gives a complete description of the component parameters.

Table 7.1 Distribution parameters for Verification Test.

Xbar	3.5000
Standard Dev.	0.2333E-01
Skewness	-1.900
Kurtosis	11.000
Reject Total %	0.236
Upper Limit %	0
Lower Limit %	0.236

To calculate the reject percentage the tolerance limits were set at ±0.093 which corresponds to a rejection fraction of 0.2361%. The system was modeled using several different batch sizes ranging from 100 to 400,000.

7.1.1 Standard Deviation

The smallest sample size simulation was 100. With this sample size the ODS method gave surprisingly close results to those of the traditional Monte Carlo. The results of the this test produced a standard deviation of 0.0199838 for the ODS method, while Monte Carlo gave a value of 0.0191037. This is only a 4.61% difference. It is also important to mention that several test were run using the 100 sample size, and while the error in the standard deviation was always quite low, the distribution approximation was random, resulting in a wide variety of distribution shapes. For this reason the results of the 100 sample size case were not includes in this section.

For the larger sample sizes, all the distributions had approximately the same variance, however, they differed significantly in the higher moments. From Tables 7.2 through 7.5 it can be seen that in general the Monte Carlo and ODS results differ slightly. The errors listed in the Tables are a result of the comparisons between the ODS method of calculation and the real system parameters. No comparison between Monte Carlo and the actual system are presented. Monte Carlo and ODS is listed. The tables show that the error between the ODS method and the real system is significantly dependant on sample size.

The next sample size to be tested was 1000 (Table 7.3). As would be expected with a larger sample size the accuracy of the ODS model increases. This test sample size proved to be more reproducible than the previous one. The results in this case produced generally the same shape distribution with a good correspondence in higher moment values.

In the final three test cases of 10,000, 100,000 and 400,00 the two methods agree quite closely. As the sample size increases, the error from the original data gets progressively smaller. The error in the mean and standard deviation was reduced to a negligible value of (tables 7.3-5). Kurtosis, however, is still almost 7% off, indicating the sensitivity of the higher moments to sample size.

Table 7.2 1,000 Simulation Comparison

1000 Sim.	Monte Carlo	ODS	% Error
Xbar	3.4996277	3.5000000	0.0
Standard Dev.	0.2311E-01	0.2311E-01	0.943
Skewness	-1.5082227	-1.5559519	18.11
Kurtosis	6.5949201	6.6901959	39.18
Reject Total %	0	0.1	57.64
Upper Limit %	0	0	0.0
Lower Limit %	0	0.1	57.64

Table 7.3 10,000 Simulation Comparison

10000 Sim.	Monte Carlo	ODS	% Error
Xbar	3.5001857	3.5000000	0.0
Standard Dev.	0.2315E-01	0.2314E-01	0.827
Skewness	-1.8205994	-1.7963515	5.456
Kurtosis	9.5791444	9.5198661	13.45
Reject Total %	0.24	0.22	6.815
Upper Limit %	0	0	0.0
Lower Limit %	0.24	0.22	6.815

Table 7.4 100,000 Simulation Comparison

100000 Sim.	Monte Carlo	ODS	% Error
Xbar	3.5000213	3.5000000	0.0
Standard Dev.	0.2326E-01	0.2326E-01	0.313
Skewness	-1.8481269	-1.8463522	2.824
Kurtosis	10.195154	10.190776	7.357
Reject Total %	0.235	0.227	3.854
Upper Limit %	0	0	0.0
Lower Limit %	0.235	0.227	3.854

Table 7.5 400,000 Simulation Comparison

400000 Sim.	Monte Carlo	ODS	% Error
Xbar	3.5000137	3.5000000	0.0
Standard Dev.	0.2328E-01	0.2328E-01	0.214
Skewness	-1.8491965	-1.8494532	2.661
Kurtosis	10.232166	10.225395	6.845
Reject Total %	0.236	0.234	0.881
Upper Limit %	0	0	0.0
Lower Limit %	0.236	0.234	0.881

In figure 7.1 the resulting distribution shapes for all the different sample sizes are compared graphically. This comparison demonstrated the difference in the extreme right and left of the distributions. The 100 sample size does not show the same number of assembly occurrence in the extreme right or left of the distribution, areas critical in the correct calculation of the acceptance fraction. The 1,000 and higher sample sizes clearly show that the frequency of assemblies with extreme dimensions should be higher than the smaller sample sizes indicate.

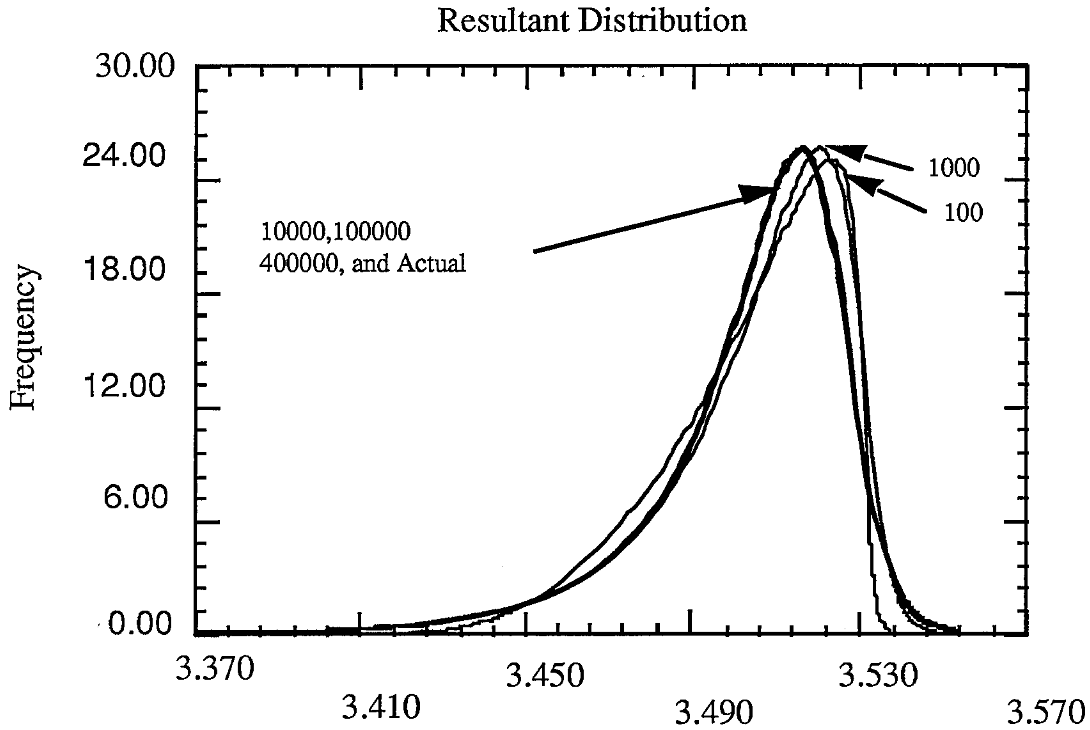


Figure 7.1 Simulated Distributions for All Sample Sizes

To compare the standard deviation of the ODS method to the real system a value of 0.23333E -1 was taken from table 7.1. The second moment appears to be less dependant on the sample size than the higher moments. For the smallest sample size of 100 the error was only 14.66%. When using

1,000 simulations this error was reduced to 1.3%. As expected the error in the standard deviation was continually reduced to 1.1% and 0.91% for the 10,000 and 100,000 sample cases respectively. Figure 7.2 shows the error between the standard deviation calculations of the ODS method and the real system versus sample size, along with the other error trends produced by the ODS method.

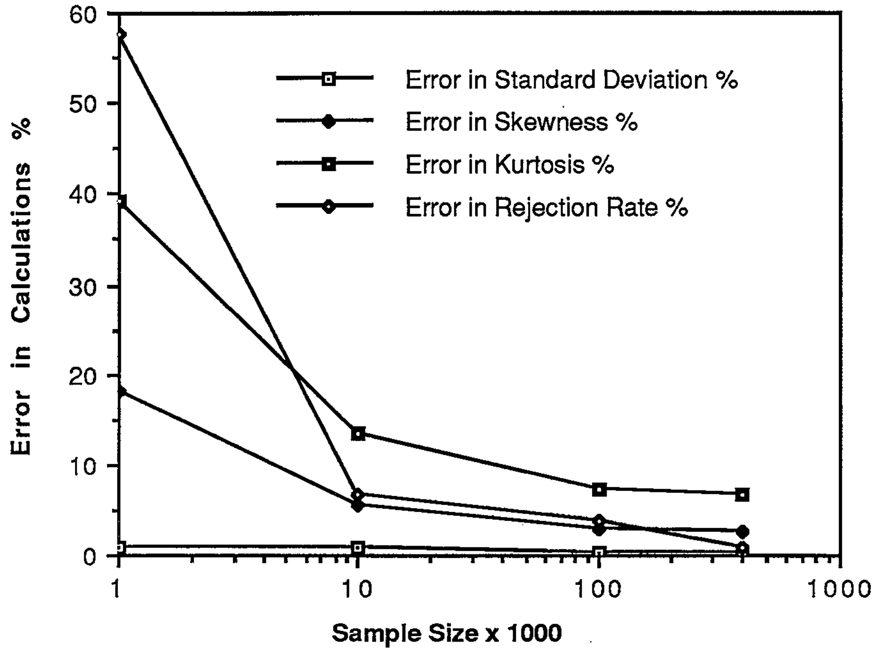


Figure 7.2 Composite Error Plots

In the following sections the higher moments, skewness and kurtosis, of the Monte Carlo simulation and the ODS method will be compared to each other and to the original system parameters.

7.1.2 Skewness

Since the higher moments seem to be the most dependant on sample size, it would be expected that the extremities of the distribution, such as the peak, the variation off the mean and the outer left and right limits, would be effected most by sample size. Since these extremities are directly related to the moments. It is clear from figure 7.1 that when using 100 samples the distribution that is generated to approximate the assembly is close to Normal. Numerically, for the 100 sample size the ODS correlates quite well with the Monte Carlo model with a difference of only 2.30%. Monte Carlo and the ODS method continued to compare quite closely for the larger sample sizes. Unfortunately, both the Monte Carlo and the ODS methods differ significantly with the real system at such a low sample size. As was shown in table 7.1, the real system has a Skewness value of -1.9. The error between the ODS method and the real system is an unacceptable 51.8%. This value improved significantly when larger sample sizes were used. The error produced for the 1,000 case was cut to 18.2%.

The approximation of the real system was dramatically improved with the larger batch sizes. The 10,000 sample size case cut the error in skewness calculations to only 4.21%. The error was reduced by almost one half to 2.7% with the 100,000 sample size. The 400,000 sample size only slightly improved the approximation to 2.6% error. From Figure 7.2 the trend in accuracy of the ODS estimation of the kurtosis as a function of sample size can be seen.

7.1.3 Kurtosis

The Kurtosis is a measure of the distribution peakedness. The system that is being estimated has a kurtosis value of 11.0. The 100 sample size simulation again produced the most inaccurate results. An error of 28.8% is produced when comparing ODS to Monte Carlo. The ODS approximation of the real systems' kurtosis is also very poor with an error of 75.8%.

As the sample size was increased the correlation with Monte Carlo continued to improve as expected. The real system approximation also continued to improve. Errors of 39.2%, 13.5%, 7.32% and 7.0% were produced when using the continually larger sample sizes.

7.1.4 Reject Percentage

The most important result of this test is the reject percentage. It is interesting to analyze the resultant distribution characteristics, but a designer is more interested in the rejects produced by a given design than the statistical characteristics of that design. To calculate the rejection rate the upper and lower tolerance limits were set at approximately the 4.5 Sigma value of $\pm$ 0.093. Since this distribution was non-Normal it is difficult to predict exactly what rejection percentage will be produced by these limits. In the analysis of the rejection fraction the correct value was obtained by using the actual system moments with equation 3.2. This equation is then solved iteratively for the percentage of the curve that falls outside of the specified tolerance limits. Using this technique, the exact rejection rate of 0.2361% for the given tolerance limits was calculated.

The first test case of one-hundred sample produced no rejects. This is expected since the tolerance limits were placed at the extremes of the distribution. These stringent tolerance requirements force simulations with small sample size to be inaccurate due to their inability to produce consistent assemblies in the extreme regions of the distribution. As was mentioned above, this sample size produced totally random results and proved to be practically useless.

In the case of 1,000 samples, there were was a predicted rejection rate of 0.1%. This differed from the correct value by 57.2%. Even though the 1,000 sample size case performs acceptably in the fitting of the distribution, it does not have the desired accuracy essential for the calculation of rejections.

The error in the rejection rate calculation is cut significantly in the 10,000 sample size and for many situations this sample size may be acceptable. However, if analysis was going to be performed on a large scale manufacturing process that produced one-million parts, the error of 6.78% would result in an miscalculation of approximately 1400 parts. When put into this perspective an apparent small error becomes noteworthy.

Finally, for the 100,000 and 400,000 sample size cases the error was reduced to 3.8% and 2.7%, respectively. For a large scale manufacturing process this error may still be much too high. At this stage it is important to note that while accuracy is very important, by increasing the sample size the time and resources invested in the analysis are increasing. A compromise must be determined by the designer between higher accuracy and cost.

At the conclusion of this section it is important to note that the distribution used for this test was highly non-Normal. Because of this many of the results were extremely sensitive to sample size. For a normal case or even a distribution that more closely approached Normal, the sample sizes would not play as important a factor as it did in this case. The use of this non-Normal distribution was done with the intent of demonstrating the importance of sample size for practical non-Normal manufacturing processes.

7.2 Assembly Size

The next major topic to consider is the effect of the number of components in the assembly. From the Central Limit theorem, it is known that as the number of parts in an assembly increase, the final assembly dimension will tend toward a central value and the distribution will tend to be Normal. This means that if an assembly has a large number of components, the resultant distribution will be tend to be Normal even if the component distributions are non-Normal. Just how much the resultant distribution deviates from Normal depends on the number of components in the assembly and the degree to which the component distributions are non-Normal. It is also affected by non-linear assembly functions.

A test was designed to determine the effects of multiple parts in an assembly. Initially the assembly consisted of a single non-Normal component. The number of components in the assembly was then increased, however, the component dimensions were scaled so that the critical assembly dimensions remained the same. By comparing these different cases, the effect that multiple parts have on the characterization of an assembly can be determined. Figure 7.3 is a representation of the one component assembly. Clearly this assembly has a highly non-Normal distribution.

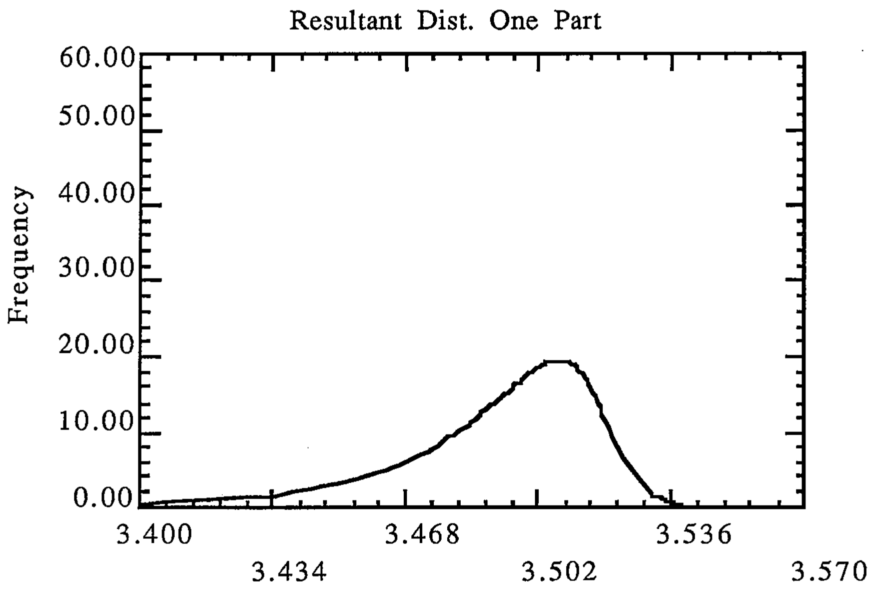


Figure 7.3 One Component Assembly

The number of components was increased by two until eight parts comprised the assembly. Figure 7.4 shows the gradual evolution of the assembly from non-Normal to an apparent Normal distribution.

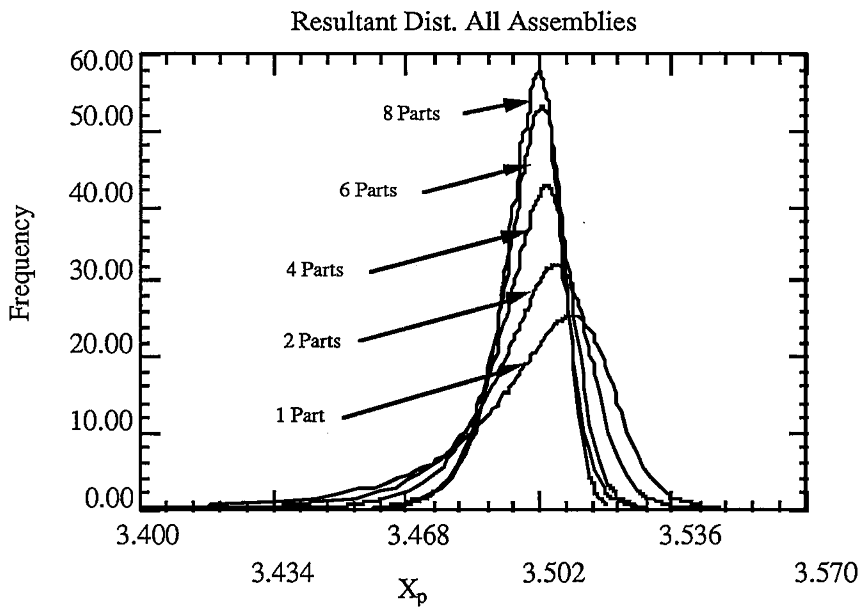


Figure 7.4 Central Tendency of a Multiple Part Assembly

Table 7.6 gives the numeric results of the moments for each specified assembly size.

Table 7.6 Moments due to Assembly Size Variation

Number of Components	Standard Deviation	Skewness	Kurtosis
1	0.02333	-1.900	11.00
2	0.01668	-1.392	7.389
4	0.01176	-1.033	5.332
6	0.00962	-0.831	4.399
8	0.00839	-0.672	3.608
Normal	----	0.000	3.000

Upon first inspection the resultant distribution of the eight part assembly looks to be Normal. To verify this, a Normal distribution is superimposed on the assembly distribution. Figure 7.5 is an enlarged view of the eight part assembly with the light gray distribution representing Normal. This figure demonstrates that while the resultant distribution is close to normal it differs most in the critical outer regions of the distribution. This region is critical because this is where most design limits are place.

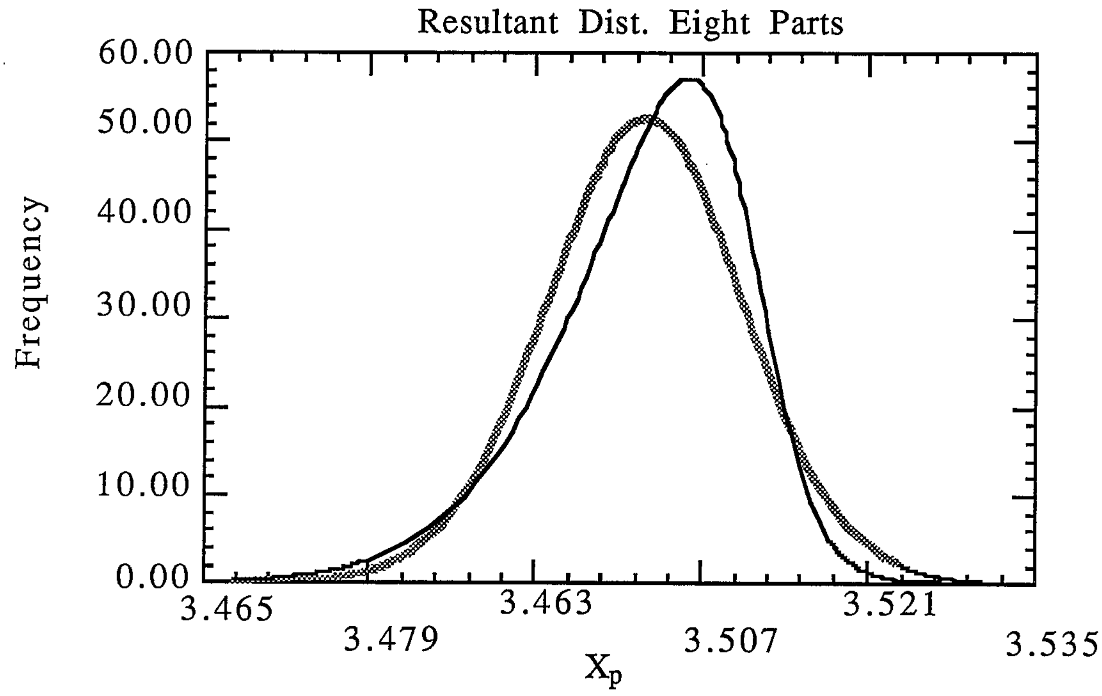


Figure 7.5 Eight Part Assembly Compared to Normal

The results of this experiment indicate that even though the assembly does tend toward a central limit and normality, the most critical regions of the distribution are affected slowly by the number of parts in the assembly. Therefore, to assume a Normal resultant distribution for assemblies with a large number of non-Normal component distributions may lead to an inaccurate analysis. Although this was an extreme example, where all components were skewed in the same direction, the designer should exercise caution.

7.3 Tolerance Allocation

The principal advantage of the ODS method is realized when applied to the problem of tolerance design or tolerance allocation. In this case, the resultant assembly tolerance limits are specified by performance requirements.

Also, the yield of the assembly process is specified, that is, the acceptance fraction.

7.3.1 Basic Concept

The available tolerance must be allocated, or distributed, among the components in such a way that the required acceptance fraction is met. This is, by nature, an iterative process. Small changes in the component tolerances are made, the resultant assembly distribution is re-computed, and the limits applied to calculate the new acceptance fraction. This process is repeated until the desired acceptance fraction is achieved. The ODS method permits iteration of the component tolerances without repeating the simulation.

It is difficult to visualize exactly what happens to the assembly distribution when a component tolerance is modified. To help in this study an assembly of three Normal components was examined using the ODS method. Upper and lower design limits were set to the three Sigma value, corresponding to a rejection rate of 0.0027. The assembly distribution and design limits are shown in figure 7.6

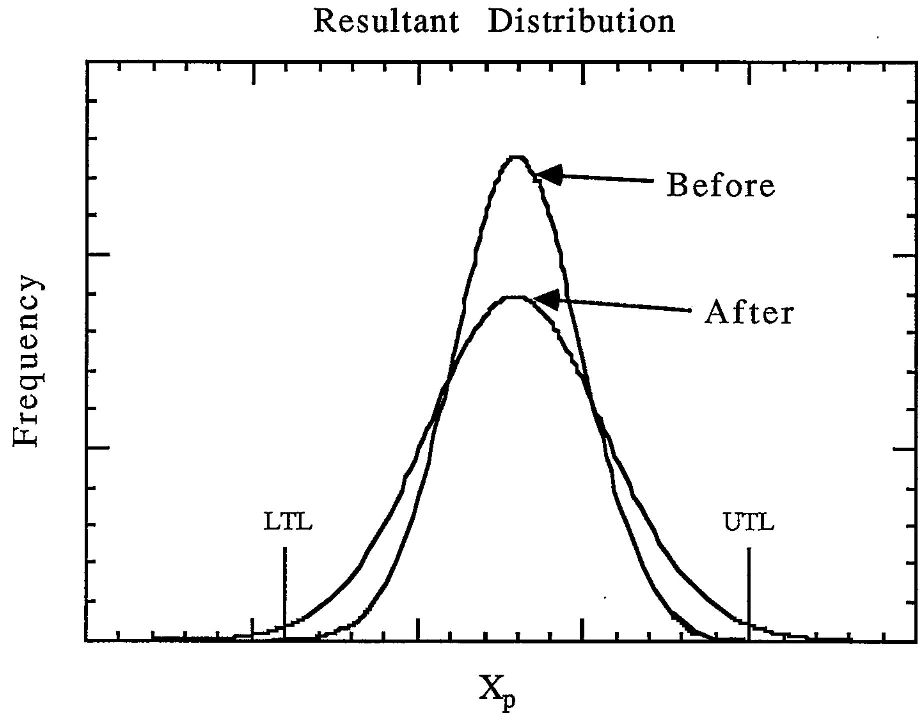


Figure 7.6 Tolerance Re-allocation of a Normal Assembly

Figure 7.6 also shows the initial assembly design as being too restrictive with a rejection rate of only 0.00012. The component tolerances were modified to be less stringent which caused the spread of the distribution to increase, and the rejection rate to rise. The shape of the distribution was only changed by a scale factor. This modification of the resultant distribution is easy to see with the Normal distribution used in this example. The following example clarifies this issue with a non-Normal case.

In this example two non-Normal components with skewness of -1.7 and kurtosis of 8.0 were assembled. The resultant assembly distribution was non-Normal as shown in figure 7.7.

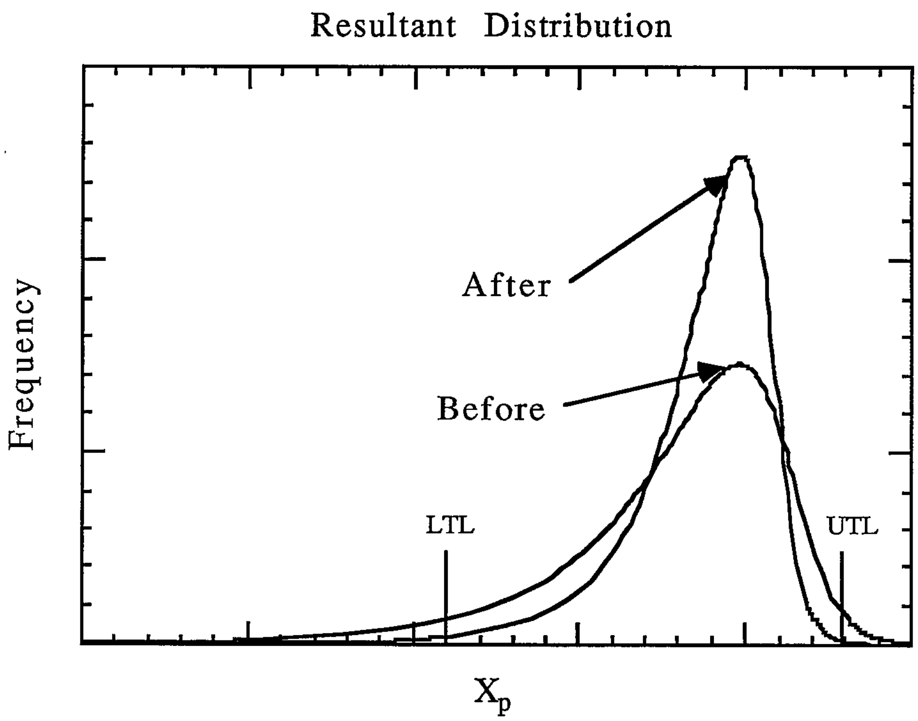


Figure 7.7 Tolerance Allocation of a Non-Normal Assembly

In this case the resulting assembly distribution was too broad, resulting in an unacceptably high rejection percentage. The component tolerances were reduced and the assembly distribution was restricted. Although the resultant distribution shape appears to be greatly modified, the skewness and kurtosis have not changed, only the standard deviation has been altered so that the rejection decreased.

7.3.2 Tolerance Allocation with ODS

Several types of allocation schemes exist that enable the designer to distribute tolerances in a logical and effective manner [2]. The original implementation of the ODS method includes a basic proportional scaling allocation algorithm. By this method, the designer initially allocated tolerances based on process capability or some other guidelines. Then, the tolerances are re-allocated by multiplying each component by a constant proportionality factor. This preserves the relative magnitude of the original component tolerances. If the the assembly tolerance is described by the root sum squared of the component tolerances (equation 7.1), the proportionality factor P can be solved for in terms of the assembly tolerance T_{Asm}. The proportionality factor is then used to scale the component tolerances T_i. The model is then re-analyzed with the new tolerances to determine the acceptability of the design

$$T_{Asm} = \sqrt{\sum P^2 T_i^2} \qquad \text{Equation 7.1}$$

This simple method of proportional scaling is not general enough to use for design. Often the designer is not free to modify all of the component tolerances. Some may be vendor supplied and not subject to change by the designer. Also, he may not know many of the statistical characteristics or their relationship to the design. Therefore, a more general approach was taken that included some important component distribution information. The more general expression of equation 7.1 is given by equation 7.2.

$$T_{Asm} = Z\sqrt{\sum P^2 \left(\frac{T_i}{Z_i}\right)^2 + \sum \left(\frac{T_f}{Z_f}\right)^2} \qquad \text{Equation 7.2}$$

Where: Z = Number of standard deviation for the assembly limits.

P = Proportionality factor.

T_i = Free component tolerances.

Z_i = Expected number of standard deviations for the free component tolerances.

T_f = Fixed Component Tolerances

Z_f = Expected number of standard deviations for the Fixed component tolerances.

Notice that only the free tolerances are scaled by the proportionality factor. This expression allows the designer to specify standard deviations corresponding to each tolerance, which may be different from the assembly limit characteristics.

The variable Z was meant to be an easy way for the designer to set the acceptability fraction for the assembly. Upon further evaluation of the ODS method, it became apparent that the requirement for the designer to know the number of standard deviations corresponding to the specified acceptance fraction for non-normal distributions was unreasonable. Since the resultant assembly distribution can be of any shape, it is impractical to assume that the designer will know the resultant distribution before hand and even more impractical to assume he will know how to represent the acceptability fraction of a non-Normal distribution by the number of its standard deviations. Therefore, the final modification on the ODS method was to allow the user to input the acceptability percentage. The software written to drive the ODS method will then convert this percentage to a Z value of the assembly distribution for use in equation 7.2. Using equation 7.2 the tolerances are then re-allocated to meet the specified acceptance percentage.

7.3.3 A Normal Example

The following example is based on the shaft housing assembly shown in figure 7.8. The purpose of this example is to compare the output of the ODS method to conventional statistical methods. This example and results of the analysis were first presented by Chase And Greenwood [2]. A shaft and housing assembly was modeled using a linear assembly function and Normal component distributions. Two tests were run using sample sizes of 10,000 and 400,000. After each simulation, the tolerances were re-allocated using the previously discussed method.

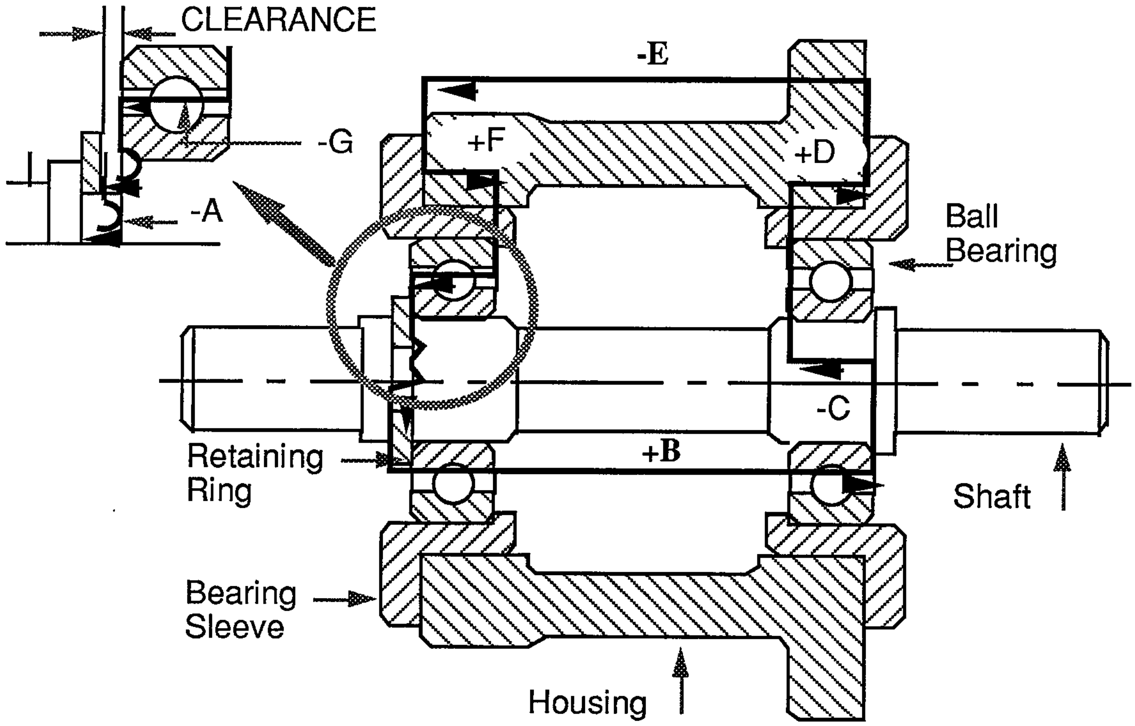


Figure 7.8 Shaft and Housing Assembly.

Using the ODS software requires the generation of an input file, with the essential tolerance and design information. Figure 7.9 is a copy of the input file for the example problem. The first line is a brief description of the problem. The second line defines the number of parts in the assembly. The next line contains the assembly specifications: the nominal assembly dimension, the tolerance limits, and the acceptance fraction, in that order. The next line contains the column heading for the component information. The last series of lines are the tolerancing and statistical information for each component.

```
THIS IS A TEST FILE FOR ODS USING SHAFT/HOUSING EXMP.
PARTS   7
ASMBLY SPECS        0.02000      -0.01500     0.01500      0.99730
PART   NOMINAL       TOLMN         TOLMX       SIG    DIST   TYP   FIXED
A      -0.0505      -0.00150     0.00150      3.00    N      0      Y
B      8.0000       -0.00800     0.00800      3.00    N      0      N
C      -0.5093      -0.00250     0.00250      3.00    N      0      Y
D      0.4000       -0.00200     0.00200      3.00    N      0      N
E      -7.711       -0.00600     0.00600      3.00    N      0      N
F      0.4000       -0.00200     0.00200      3.00    N      0      N
G      -0.5093      -0.00250     0.00250      3.00    N      0      Y
```

Figure 7.9 ODS input File For Example Problem.

The next step in the analysis is to select an appropriate sample size. A sample size of 10,000 was used for this example. The sample input file is then read and the results of the initial design are displayed with the option to re-allocate the tolerances according to the specified acceptance fraction. The initial display shows both the ODS and Monte Carlo simulation results. If the designer chooses to modify the component tolerances a new set of only the ODS results are displayed. Figure 7.10 is a sample output of the ODS analysis.

```
-----------------------------SIMULATED-----------SPECIFIED----
  Assembly Tolerances
     Upper Assembly Tol     =     0.01104           0.01500
     Lower Assembly Tol     =     -0.01104          -0.01500

  Number of Standard Deviations
     Upper                  =     4.07565           3.00000
     Lower                  =     -4.07565          -3.00000
------------------------------------------------------------------
   Monte Carlo                              ODS
 XBAR     =  0.20960758E-01             XBR2     = 0.20000000E-01
 STDEV    =  0.37182106E-02             SDEV2    = 0.36803904E-02
 SKEW     =  -0.34380581E-01            SKEW2    = -0.68100716E-01
 KURT     =  3.0867989                  KURT2    = 3.0668279

 REJTS    =  0                          CREJT    = 0
 REJTU    =  0                          CREJU    = 0.42000000E-04
 REJTL    =  0                          CREJL    = 0.40000000E-05
```

Figure 7.10 ODS Results.

If the designer desires he can then look at the assembly distribution. Figure 7.11 is a distribution of the Shaft and Housing example problem before and after the re-allocation of tolerances with the design limits marked as UTL and LTL.

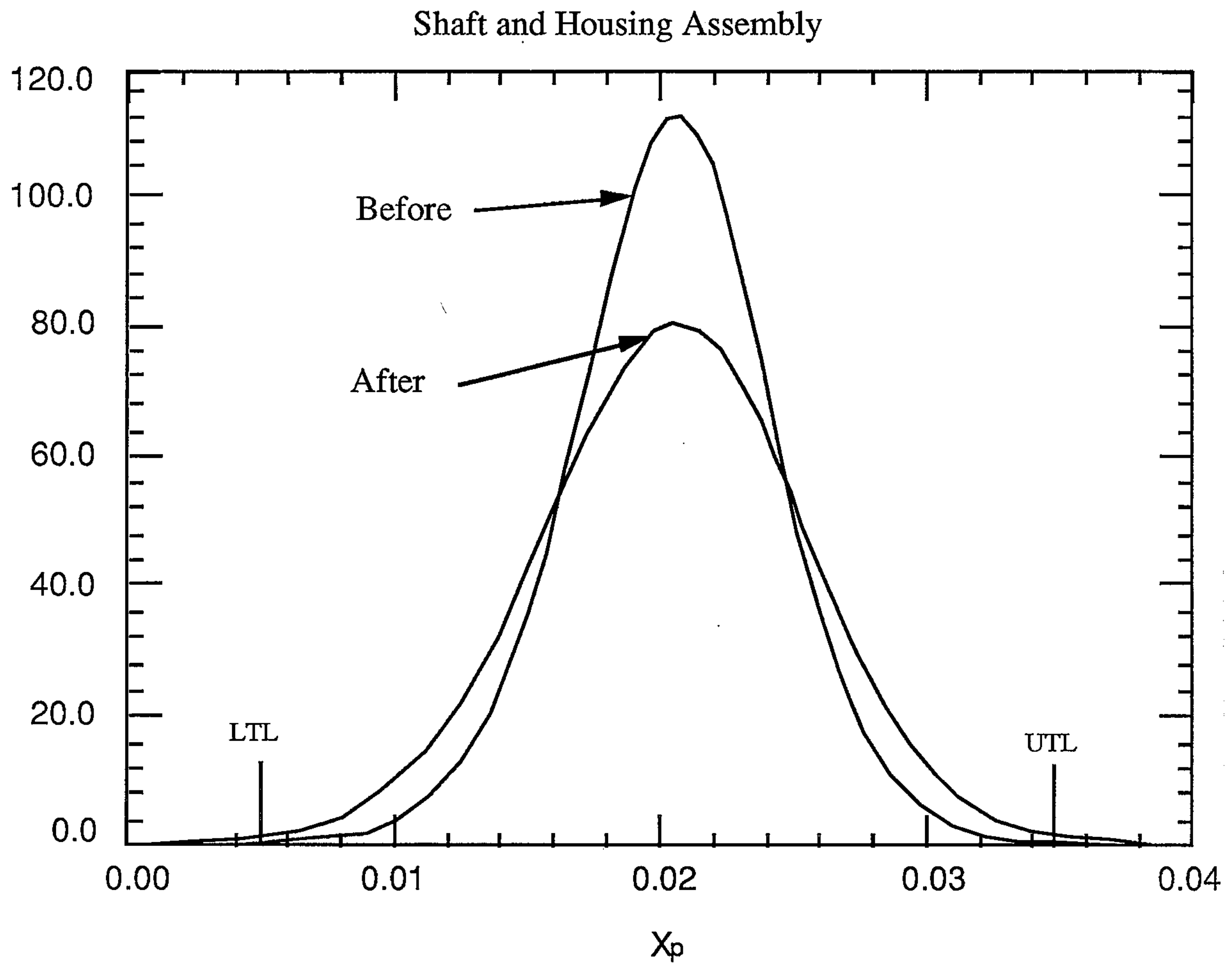


Figure 7.11 Gear and Housing Distribution.

By comparing the re-allocated tolerances produced by ODS to the text answers, verification of the ODS method can be made. The results generated by the ODS method appear in Table 7.7 along with tolerances allocated by two other statistical methods [2]. The values designated with an asterisk are designated as vendor-supplied and must be fixed when allocation methods are used.

Table 7.7 Allocation Comparison

Part	Nominal Dimension	Original Tolerance	ODS Method 10000	ODS Method 400000	RSS Proportional Scaling
A	0.0505	0.0015*	0.0015*	0.0015*	0.0015*
B	8.0000	0.008	0.01071	0.01118	0.01116
C	0.5093	0.0025*	0.0025*	0.0025*	0.0025*
D	0.4000	0.002	0.00268	0.00280	0.00279
E	7.711	0.006	0.00803	0.00839	0.00837
F	0.4000	0.002	0.00268	0.00280	0.00279
G	0.5093	0.0025*	0.0025*	0.0025*	0.0025*

* Fixed tolerances

It is important to notice the difference in results due to the sample size. Since the sample problem is a linear assembly of Normal components, the straight statistical results are exact. ODS was accurate within 5% using 10,000 samples, however, it was within 0.2% for 400,000 samples. This is a significant result since designers using Monte Carlo often assume that smaller sample sizes are sufficient.

It is clear from these results that the ODS method does accurately permit allocation of tolerances, without repeating the simulation, provided the sample size is sufficient to give good distribution shape information. However, this example only tested the case of a Normal distribution and a linear assembly function.

7.3.4 A Non-Normal Example

The final test of the ODS method was to allocate tolerances for a non-Normal set of components. For this test the same assembly of the shaft and housing from the previous test was used, except that the component distributions were changed from Normal to skewed. ODS was then used to allocate the variable tolerances. The resultant assembly was then evaluated with traditional Monte Carlo. Since it was determined that Monte Carlo is accurate for large sample sizes, the effectiveness of the ODS method was verified by comparing the yield specified in the ODS simulation to the results of the Monte Carlo simulation.

An assembly acceptance fraction of 99.77% was specified for the ODS simulation. The modified proportional scaling method in ODS adjusted the five variable tolerances. Using these adjusted tolerance values in the Monte Carlo simulation produced an acceptance fraction of 99.78%. This corresponds to an error of 5% in the number of rejects. This is compared to the magnitude of the error for the 200,000 sample size employed. Table 7.8 gives a summery of the results for the non-Normal case.

Table 7.8 Non-Normal Allocation Results

	ODS Method 200000	Monte Carlo	% Error
Mean	0.0209	0.0210	0.476
Standard Dev.	0.387E-2	0.387E-2	0.007
Skewness	0.7206	0.7061	2.053
Kurtosis	3.7374	3.7079	0.810
Rejects	460	440	4.50
Acc. Fraction	99.77	99.78	0.01

Part	Allocated Tolerances	
A	0.0015*	
B	0.00845	
C	0.0025*	
D	0.00211	
E	0.00634	
F	0.00211	
G	0.0025*	

Figure 7.12 shows a comparison of the distributions produced by ODS and Monte Carlo based on the specified acceptance fraction.

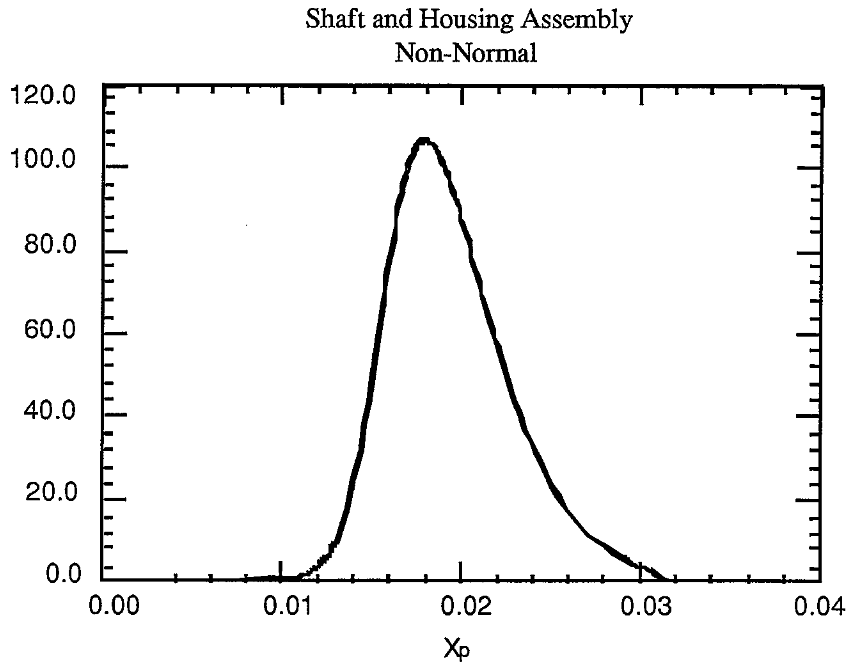


Figure 7.12 Non-Normal Tolerance Allocation

Based on these results, it is concluded that the ODS method does accurately allocate tolerances for both Normal and non-Normal assemblies.

8. Conclusion and Recommendations

This section is a summary of all the results of this thesis. It includes the conclusions drawn from the research that was performed as well as some suggestions for further research in this area.

8.1 Contributions

A new and efficient method of designing by tolerance allocation was developed. This method, called Optimal Design Simulation (ODS), is particularly suited for design iteration. It is based on the storing of compact statistical data for the assembly distribution in parametric form. Based on this statistical data, the component tolerances can be re-allocated and re-evaluated until an acceptable design is achieved.

Traditional Monte Carlo was analyzed for dependencies on sample size and the number of components in an assembly. Results based on this analysis help to determine sample size requirements needed for accurate tolerance analysis.

A Hybrid driver using the Method of System Moments and Monte Carlo techniques was also developed to determine the effectiveness of this concept in reducing the amount of computer time used for traditional Monte Carlo. This driver also served as a foundation for the ODS Method

A versatile random variate generation algorithm was developed based on the Generalized Lambda Family of distributions. This method of generation allows the approximation of a wide variety of distributions including uniform, Normal, and wide range of non-Normal.

8.2 Conclusions

8.2.1 Random Number Generation

The generation of random variates has been over-simplified in many cases. While at first glance it seems almost trivial, there are some subtle issues that must be evaluated. Both the number of non-repeating random variates produced by a generator and the uniformity of the distribution are critical to the accuracy of the simulation. The effect of the random number generator was not tested directly in this thesis, but in the single component test it was demonstrated that the the same distribution was generated as was input, at least to sufficient accuracy for comparing acceptance fraction to four significant figures.

The Generalized Lambda Method of number generation appears to be an effective method to apply with the statistical techniques used in this research. It allows the generation of a wide range of distribution shapes with one equation. Using this equation in an iterative manner, the rejection percentage of the assembly can be found quickly and accurately. Using this method of variate generation also assisted in conversion from acceptance percentage to standardized variates, vital in the re-allocation scheme developed in this thesis.

8.2.2 Traditional Monte Carlo

Monte Carlo Simulation is an effective tool when used for tolerance analysis. It is conceptually simple and it effectively simulates assembly problems intrinsic to most manufacturing process. Monte Carlo is ideal for non-Normal component distributions and non-linear assembly functions.

Unfortunately, for all the benefits of Monte Carlo the price of computer time and resources must be paid. Monte Carlo only provides accurate results for large sample sizes, therefore, to use traditional Monte Carlo as an iterative design tool is impractical.

8.2.3 Method of system Moments

The Method of System Moments is an alternative to Monte Carlo simulation. The computational time used in this method is significantly less than that used in Monte Carlo. The draw back to this method is the difficulty encountered when using non-linear assembly functions. Since the method is based on second order partial derivatives or higher, problems with a non-linear assembly functions become very difficult.

8.2.4 The Hybrid Method

The Hybrid method tries to take advantage of both Monte Carlo Simulation and the Method of System Moments. It was originally developed to reduce the computer time used in Monte Carlo by using the system moments in conjunction with random simulation of the assembly. The results in Chapter 7 show that this is generally not true for most distributions. It appears to have the same dependence on sample size as traditional Monte Carlo.

Even though the initial intent of the Hybrid method was not successful, the concept is essential in the generation of a the new Optimum Design Simulation method.

8.2.5 Optimum Design Simulation

Optimum Design Simulation (ODS) is a powerful new tool used for iterative tolerance analysis and design. When used as a design tool, ODS allows the accurate simulation of non-Normal component distributions and iterative design of assembly processes in a fraction of the computer time taken by Monte Carlo. It is not, however, a one time simulation tool. If used to simulate only one set of design specifications without iteration, it is more computer intense than Monte Carlo. This is primarily due to the time taken to store component information. The benefit of ODS comes when used for iterative design work. By recording the distribution characteristics ODS can re-simulate without having to generate the distribution in future simulations.

8.5.2.1 Allocation Capability

The tolerance allocation capability of the ODS method has been demonstrated for several one-dimensional assemblies. Both Normal and non-Normal component distributions were analyzed. The re-allocated tolerances were precisely meet the prescribed assembly acceptance requirements to four significant digits of accuracy and predict rejects to 5% accuracy. Therefore, it can be concluded that the ODS method of design iteration can achieve the same accuracy as the Monte Carlo method without the expense of repeating the simulation.

The only allocation algorithm demonstrated was proportional scaling, but it is felt that a full rang of allocation techniques should work equally as well, including cost driven optimization methods.

No tests were done exploring the effects of tolerance allocation for non-linear assembly functions (2-D or 3-D assemblies). The derivative of the system moments included sensitivities, and they have been included in the computer implementation. Therefore, nonlinear assembly functions can be analyzed and tolerances may be allocated in the same way by the ODS method. However, since only the first order derivative terms were included in the parameterized moment expression, the full nonlinear effects will not be taken into account. A nonlinear assembly function can produce an asymmetric assembly distribution even though all the components distributions may be symmetric. The linearized assembly function model used in the ODS moment expression would not describe such second order effects.

8.5.2.2 Sample Size

The sample size used with the ODS methods is dependant on the degree of accuracy desired in the approximation of the manufacturing process and the Normality of that process. ODS requires a sample size approximately the same magnitude as that of Monte Carlo simulation. Based on the tests performed, it appears that the more Normal the resulting assembly is the less simulations are required to approximate the process distribution shape accurately. However, to attain the accuracy needed for acceptance rate calculation, simulations on the order of four hundred thousand must be performed. This is primarily due to the small values encountered at the extremes of the distribution. The extreme tails of the distribution is where tolerance re-allocation has the greatest effects.

8.5.2.3 Multiple Part Assemblies

As the number of parts in the assembly increases the final dimension of the assembly tends toward a central limit. Based on this fact many designers assume that most assemblies can be correctly modeled by a Normal distribution. This is an incorrect assumption. While it is true that the assembly does tend toward a central limit, this does not mean that the distribution is Normal. This is particularly evident in the extremes of the distribution. Designers should be most concerned about the extremes of the distribution since this is where tolerance limits are specified and most re-allocation effects are seen. Therefore, by oversimplifying the problem and assuming a Normal resultant distribution, inaccurate results are generated.

8.3 Recommendations

The generation of random variates became a very important topic in this thesis. It is clear that future research in this area must be done to determine the effects of random number generation on various simulation techniques.

After a simulation is completed, the resultant distribution is approximated by the Generalized Lambda equation. The four lambda parameters are obtained from a table using an interpolation scheme. Therefore, the fitting possibilities for a resultant distribution are limited by the range of the table. It would be desirable to derive a general method of determining the Lambda parameters. This method may also be more accurate in the approximation of the distribution.

The re-calculation of the assembly distribution from the parameterized distribution moments was based on a first order approximation of the assembly function. This is a linear technique and therefore limited to an approximation for non-linear problems. The accuracy of this method for re-allocated tolerances that deviate greatly from the initial tolerances should be evaluated. It would also be interesting to determine which tolerances were contributing most to the variation in the assembly. This concept has been discussed by Rubenstein [15].

ODS was evaluated and tested using only one-dimensional tolerance loops. Sensitivities have been included in the ODS software that allow for multiple dimension analysis, however, no evaluation was performed. This is due primarily to the time involved with the changing of the assembly function. In the current ODS software the assembly function is hard coded in the program. A general method of entering the assembly function would be an asset to this software.

When the tolerances were uncoupled from the system parameters in Chapter 5, the assumption was made that the system moments of the resulting distribution could be expressed linearly. This approximation proved to effectively simulate many manufacturing processes accurately, however there are some non-linear effects that are neglected. These effects should be studied and it should be determined if the linearization causes any significant problem.

References

[1] Burr, Irving W., "Tolerances for Mating Parts", Industrial Quality Control, September 1958, pp. 18-22.

[2] Chase, Dr. K. and W. H. Greenwood, "Design Issues in Mechanical Tolerance Analysis" ADCATS, Report No. 87-5.

[3] Corlew, Gordon T. and Fred Oakland, "Monte Carlo Simulation for Setting Dimentional Tolerances", Machine Design, May 6, 1976, pp. 91-95.

[4] Cox, Niel D. ,"How to Perform Tolerance Analysis", ASQC Basic Refrences in Quality Control: Statistical Techniques, Vol. 11, ASQC, 1986.

[5] DeDoncker, Dwight and Al Spencer, "Assembly Tolerance Analysis with Simulation and Optimization Techniques", SAE Technical Paper Series, Paper number 870263, Febuary 1987.

[6] Evans, D.H., "Statistical Tolerancing: The State of the Art, Part 1." Journal of Quality Technology, Vol. 6, No. 4, Oct. 1974, pp. 188-195. Part 2., vol. 7, no. 1, Jan. 1975, pp.1-12.

[7] Grossman, David D., "Monte Carlo Simulation of Tolerancing in Discrete Parts Manufacturing and Assembly" Stanford Artificial Intelligence Laboratory, Computer Science Department Report Number STAN-CS-76-555, May 1976.

[8] Greenwood, W. H., "A New Tolerance Analysis Method for Engineering Design and Manufacturing", Ph.D. dissertation, Mechanical Engineering Department, Brigham Young University, March, 1987.

[9] Hahn, G.J. and S.S Shapiro, Statistical Models in Engineering, John Wiley & Sons, Inc., New York, 1967.

[10] Held, David O. , "Assembly Variation Analysis: Liberty's Process driven Design Tool for Quality", SAE Technical Paper Series, Paper Number 880606, March 1988.

[11] Lehtihet, E.A. and B.A. Dindelli, "TOLCON: Microcomputer Based Module for Simulation of Tolerances", Department of Industrial & Management Systems Engineering, Pennsylvania State University.

[12] Press, William H., Brian P. Flannery, Saul A. Teukoslky and WIlliam T. Vetterling, Numerical Recipies, Cambridge University Press, New York,1986.

[13] Ragon, Dan and Al Spencer, "John Deere Assembly Variation of Crawler Track Chain/Shoe Assembly", SAE Technical Paper Series, Paper Number 881243, September 1988.

[14] Ramberg, J.S., E.J Dudewicz, P.R. Tadikamalla and E.F. Mykytka, "A Probobility Distribution and Its Uses in Fitting Data", Technometrics©, Vol. 21, No. 2, May 1979, pp 201 -209.

[15] Rubinstein, Reuven Y., Monte Carlo Optimization, Simulation and Sensitivity of Queueing Networks, John Wiley & Sons, Inc., New York, 1986.

[16] Shapiro, S.S. and A.J. Gross, Statistical Modeling Techniques, Marcel Dekker, Inc., 1981.

[17] Spotts, M.F., Dimensioning and Toleranceing for Quantity Production, Prentice-Hall Inc.,Englewood Cliffs, N.J.,1983.

[18] Stoodley, K.D.C., T. Lewis and C.L.S. Stainton, Applied Statistical Techniques , John Wiley & Sons, Inc., New York, 1980.

[19] Stoodley, K.D.C., Applied and Computational Statistics, John Wiley & Sons, Inc., New York, 1984.

[20] Velsher, B., "Tolerance Analysis in Electronic Packaging", Electronic Packaging & Production, April 1987,pp. 76-78.

[21] Whitlock, Kalos, Monte Carlo Methods, John Wiley & Sons, Inc., New York, 1986.

Appendix A

The FORTRAN code that implements the ODS method is in a file called ODS.for. This section contains a calls map as well as a brief description of each of the important subroutine modules.

Calls Map

PROGRAM ODS
- Subroutine LIB$ERASE_PAGE
- Function IENTER
  - Fountain QPRMT
- Function FUPCAS
- Function RANO
- Subroutine LAMDA
  - Subroutine LTABLE
- Subroutine COFSQR
- Subroutine COFCUB
- Subroutine COFQAD
- Subroutine DUMP
- Subroutine TRASQR
- Subroutine TRACUB
- Subroutine TRAQAD
- Subroutine LTABLE
- Subroutine REJECT
- Subroutine TEKGPH

Program Subroutine Descriptions

Subroutine LIB$ERASE_PAGE (1,1)

Description : VAX system call to clear the screen.

Function IENTER

Description : IENTER is a generic user input routine.

Function FUPCAS

Description : FUPCAS converts a string to all upper case.

Function RANO

Description : This function enhances the VAX system random generator.

Subroutine LAMDA(NUMDIS,RLAM,B1,B2,BLENTH,PTOLMX,PTOLMN,PSIG)

Description : LAMDA uses the inputs B1 and B2 to determine a distribution and return the shape parameters in the array RLAM.

Subroutine COFSQR(IPARTS,TEMP,ALFASQ,IOFFSQ)

Description : COFSQR is the routine used by ODS to separate and store all of the coefficients for the second moments. These values are stored in ALFASQ.

Subroutine COFCUB(IPARTS,TEMP,ALFACB,IOFFCB)

Description : COFCUB like COFSQR is used to separate and store all of the coefficients. This routine is used for the third moments. These values are stored in ALFACB.

Subroutine COFQAD(IPARTS,TEMP,ALFAQD,IOFFQD)

Description : COFQAD separates and store all of the coefficients for the fourth moment in ALFAQD.

Subroutine DUMP(ALFASQ,210,'ALFASQ')

Description : Is a debugging routine used to dump out the contents of either ALFASQ,ALFACB, or ALFAQD..

Subroutine TRASQR(IPARTS,ALFASQ,IOFFSQ,PTLRNG,SUMSQ)

Description : TRASQR is used to translate the stored data in ALFASQ into a value (SUMSQ) that can be used to calculate the second moment.

Subroutine TRACUB(IPARTS,ALFACB,IOFFCB,PTLRNG,SUMCB)

Description : TRACUB is used to translate the stored data in ALFACB into a value that can be used to calculate the third moment.

Subroutine TRAQAD(IPARTS,ALFAQD,IOFFQD,PTLRNG,SUMQD)

Description : TRAQAD is used to translate the stored data in ALFAQD into a value that can be used to calculate the fourth moment.

Subroutine LTABLE(ABS(B1),ABS(B2),RLAM,1,IERR)

Description : LTABLE contains all the shape information presented in the Lambda tables. It performs an interpolation bases on the inputs B1 and B2. It then returns the four lambda parameters in RLAM.

Subroutine REJECT(PLAM,ANOM+TLU,ANOM+TLL,CREJU,CREJD)

Description : This routine used iteration the generalized lambda equation to determine the percent of rejects..

Subroutine TEKGPH(IOUT,INP,X,Y,ICURVE,NPTS,MXPT,LTIT,LBLH,LBLV)

Description : This is the graphics driver used to display the distributions.

Appendix B

Generalized Lambda Table
Alfa3 = 0.10 Lam1 -1.678,-1.271,-0.872,-0.515,-0.269,-0.164,-0.117,
-0.092,-0.076,-0.065,-0.057,-0.049,-0.048,-0.046,-0.044,-0.041,-0.037,
-0.036,-0.033,-0.032,-0.030,-0.028,-0.027,-0.027,-0.025,-0.024,-0.023,
-0.023,-0.022,-0.021,-0.020,-0.020,-0.020,-0.019,-0.019,-0.018,-0.017,
-0.017,-0.017
Alfa3 = 0.10 Lam2
0.2835,0.3028,0.3177,0.3164,0.2863,0.2417,0.1977,0.1572,0.1203,0.0866,
0.0558,0.0276,0.0142,0.001440,-0.0109,-0.0227,-0.0452,-0.0661,-0.0857,
-0.104,-0.1213,-0.1375,-0.15530,-0.1674,-0.1811.-0.1943,-0.2066,-0.2184,
-0.2297,-0.2405,-0.2507,-0.2606,-0.2699,-0.2791,-0.2878,-0.2961,-0.3041,
-0.3191,-0.3193
Alfa3 = 0.10 Lam3
0.00,0.0412,0.0941,0.1477,0.1678,0.1486,0.1205,0.0936,0.0698,0.0490,0.030
8,0.0149,0.007606,0.000762,-0.005703,-0.0118,-0.0231,-0.0332,-0.0424,
-0.0507,-0.0584,-0.0654,-0.0719,-0.778,-0.0834,-0.0886,-0.0934,-0.0979,
-0.1021,-0.1061,-0.1099,-0.1134,-0.1167,-0.1199,-0.1229,-0.1258,-0.1285 ,
-0.1311,-0.1335
Alfa3 = 0.10 Lam4
0.9070,0.7373,0.570,0.4116,0.2831,0.2033,0.1503,0.1111,0.0803,0.0552,
0.0342,0.0163,0.008302,0.000828,-0.006174,-0.0127,-0.0247,-0.0354,
-0.0450,-0.0537,-0.0616,-0.0688,-0.0755,-0.0816,-0.0872,-0.0925,-0.0973,
-0.1019,-0.1062,-0.1102,-0.1139,-0.1175,-0.1208,-0.1240,-0.1270,-0.1298,
-0.1325,-0.1351,-0.1376
Alfa3 = 0.15 Lam1
-1.655,-1.323,-0.940,-0.617,-0.376,-0.244,-0.177,-0.138,-0.114,-0.098,
-0.086,-0.076,-0.073,-0.069,-0.066,-0.063,-0.058,-0.055,-0.051,-0.048,
-0.045,-0.043,-0.042,-0.040,-0.038,-0.037,-0.035,-0.034,-0.033,-0.032,
-0.031,-0.030,-0.029,-0.028,-0.028,-0.027,-0.027,-0.026,-0.025
Alfa3 = 0.15 Lam2
0.2811,0.2934,0.3056,0.3031,0.2791,0.2397,0.1980,0.1584,0.1219,0.0884,
0.0577,0.0294,0.0160,0.003217,-0.009113,-0.021,-0.0435,-0.0644,-0.0842,
-0.1025,-0.1198,-0.1356,-0.1514,-0.1660,-0.1798,-0.1928,-0.2053,-0.2172,
-0.2284,-0.2392,-0.2496,-0.2593,-0.2688,-0.2780,-0.2866,-0.2948,-0.3031,
-0.3108,-0.3183
Alfa3 = 0.15 Lam3
0.0000,0.0314,0.0782,0.1215,0.1435,0.1350,0.1355,0.0901,0.682,0.0486,
0.0310,0.0155,0.008378,0.001667,-0.004680,-0.0107,-0.0218,-0.0318,
-0.0410,-0.0493,-0.0569,-0.0639,-0.0703,-0.0763,-0.0819,-0.0870,-0.0919,
-0.0964,-0.1006,-0.1046,-0.1084,-0.1119,-0.1153,-0.1185,-0.1215,-0.1243,
-0.1271,-0.1297,-0.1322
Alfa3 = 0.15 Lam4
0.87,0.7204,0.5623,0.4194,0.2994,0.2156,0.1586,0.1167,0.0843,0.0581,
0.0363,0.0178,0.009564,0.001890,-0.005278,-0.0120,-0.0242,-0.0351,

-0.0449,-0.0537,-0.0617,-0.0690,-0.0757,-0.0819,-0.0876,-0.0929,-0.0978,
-0.1024,-0.1067,-0.1107,-0.1145,-0.1180,-0.1214,-0.1246,-0.1276,-0.1304,
-0.1332,-0.1357,-0.1382
Alfa3 = 0.20 Lam1
-1.387,-1.011,-0.706,-0.471,-0.322,-0.237,-0.187,-0.154,-0.132,-0.116,
-0.103,-0.097,-0.093,-0.089,-0.079,-0.074,-0.069,-0.065,-0.061,-0.058,
-0.055,-0.053,-0.051,-0.049,-0.047,-0.045,-0.044,-0.043,-0.041,-0.040,
-0.039,-0.038,-0.037,-0.036,0.035,-0.035,-0.034,-0.034
Alfa3 = 0.20 Lam2
0.2841,0.0947,0.2919,0.2718,0.2374,0.1983,0.1599,0.1240,0.0908,0.0601,
0.0318,0.0185,0.005707,-0.006641,-0.0185,-0.0410,-0.0622,-0.0818,
-0.1003,-0.1176,-0.1339,-0.1494,-0.1639,-0.1778,-0.1909,-0.2034,-0.2153,
-0.2265,-0.2374,-0.2477,-0.2577,-0.2671,-0.2762,-0.2850,-0.2935,-0.3014,
-0.3092,-0.3168,-0.3241
Alfa3 = 0.20 Lam3
0.0212,0.0638,0.1013,0.1233,0.1221,0.1065,0.0866,0.0667,0.0482,0.0314,
0.0164,0.009467,0.002894,-0.003342,-0.009261,-0.0202,-0.0302,-0.392,
-0.0475,-0.0551,-0.0621,-0.0686,-0.0745,-0.0801,-0.0853,-0.0901,-0.0947,
-0.0989,-0.1029,-0.1067,-0.1103,-0.1136,-0.1168,-0.1199,-0.1228,-0.1255,
-0.281,-0.1306,-0.1330
Alfa3 = 0.20 Lam4
0.7090,0.5571,0.4246,0.3120,0.2273,0.1672,0.1230,0.0889,0.615,0.0389,
0.0198,0.0113,0.003429,-0.003929,-0.0108,-0.0233,-0.0345,-0.0444,
-0.0534,-0.0615,-0.0689,-0.0757,-0.0819,-0.0877,-0.0930,-0.0980,-0.026,
-0.1069,-0.1110,-0.1148,-0.1184,-0.1218,-0.1250,-0.1280,-0.1309,-0.1336,
-0.1362,-0.1387,-0.1411
Alfa3 = 0.25 Lam1
-1.465,-1.084,-0.790,-0.558,-0.398,-0.298,-0.237,-0.196,-0.167,-0.147,
-0.131,-0.126,-0.118,-0.113,-0.108,-0.099,-0.094,-0.087,-0.082,-0.077,
-0.073,-0.070,-0.067,-0.064,-0.062,-0.059,-0.058,-0.055,-0.054,-0.052,
-0.051,-0.049,-0.048,-0.047,-0.046,-0.044,-0.044,-0.043,-0.042
Alfa3 = 0.20 Lam2
0.2748,0.2847,0.2820,0.2650,0.2349,0.1987,0.1619,0.1266,0.0937,0.0632,
0.0351,0.0217,0.008889,-0.003476,-0.0154,-0.0380,-0.0591,-0.790,-0.0974,
-0.1149,-0.1312,-0.1467,-0.1613,-0.1753,-0.1885,-0.2010,-0.2129,-0.2242,
-0.2350,-0.2455,-0.2554,-0.2649,-0.2742,-0.2829,-0.2914,-0.2995,-0.3072,
-0.3147,-0.3220
Alfa3 = 0.20 Lam3
0.105,0.0506,0.0843,0.1062,0.1099,0.0996,0.0831,0.0653,0.481,0.321,
0.0176,0.108,0.004408,-0.001713,-0.007540,-0.0184,-0.0282,-0.373,
-0.0455,-0.0531,-0.0601,-0.0665,-0.0725,-0.0781,-0.0833,-0.0882,-0.0927,
-0.0970,-0.1010,-0.1048,-0.1084,-0.1118,-0.1151,-0.1181,-0.1210,-0.1238,
-0.1264,-0.1289,-0.1313
Alfa3 = 0.20 Lam4
0.7034,0.5548,0.4294,0.3226,0.2385,0.1763,0.1300,0.0942,0.0656,0.0421,
0.0224,0.0136,0.005467,-0.002103,-0.009175,-0.0220,-0.0334,-0.0436,
-0.0527,-0.0610,-0.0685,-0.0754,-0.0817,-0.0876,-0.0930,-0.0980,-0.1027,
-0.1070,-0.1111,-0.1150,-0.1186,-0.1220,-0.1252,-0.1283,-0.1312,-0.1339,
-0.1365,-0.1390,-0.1414
Alfa3 = 0.30 Lam4
0.7020,0.5556,0.4348,0.3324,0.2495,0.1859,0.1377,0.1003,0.704,0.460,
0.0255,0.008035,0.00489,-0.007057,-0.0139,-0.0203,-0.0319,-0.0423,
-0.0517,-0.601,-0.678,-0.0748,-0.0812,-0.0872,-0.0927,-0.977,-0.1025,
-0.1069,-0.1111,-0.1149,-0.1186,-0.1220,-0.1253,-0.1284,-0.1313,-0.1341,

-0.1367,-0.1392,-0.1416
Alfa3 = 0.35 Lam2
0.2639,0.2668,0.2653,0.2528,0.2298,0.1996,0.1665,0.1333,0.1014,0.0714,
0.0434,0.0173,0.004870,-0.007105,-0.0187,-0.0298,-0.0511,-0.0710,
-0.0898,-0.1074,-0.1240,-0.1396,-0.1545,-0.1685,-0.1818,-0.1945,-0.2067,
-0.2181,-0.2291,-0.2396,-0.2496,-0.2593,-0.2685,-0.2775,-0.2860,-0.2942,
-0.3020,-0.3096,-0.3172
Alfa3 = 0.35 Lam3
0.0000,0.0256,0.0559,0.0775,0.0873,0.0854.,.0758,0.0625,0.0482,0.0340,
0.0206,0.008158,0.002293,-0.003332,-0.008723,-0.0139,-0.0236,-0.0325,
-0.0407,-0.0483,-0.0553,-0.0618,-0.0678,-0.0735,-0.0787,-0.0836,-0.0883,
-0.0926,-0.0967,-0.1006,-0.1042,-0.1077,-0.1109,-0.1141,-0.1170,-0.1198,
-0.1225,-0.1251,-0.1276
Alfa3 = 0.35 Lam4
0.6836,0.5599,0.4415,0.3423,0.2606,0.1961,0.1462,0.1072,0.0760,0.505,
0.0293,0.0112,0.003090,-0.004431,-0.0115,-0.0180,-0.0300,-0.0407,
-0.0503,-0.0589,-0.0668,-0.0739,-0.0805,-0.0865,-0.0921,-0.0973,-0.1021,
-0.1066,-0.1108,-0.1147,-0.1184,-0.1219,-0.1252,-0.1283,-0.1313,-0.1341,
-0.1367,-0.1392,-0.1417
Alfa3 = 0.40 Lam1
-1.354,-.043,-0.808,-0.627,-0.494,-0.400,-0.333,-0.284,-0.248,-0.222,
-0.200,-0.190,-0.182,-0.174,-0.166,-0.155,-0.146,-0.136,-0.129,-0.122,
-0.115,-0.111,-0.106,-0.102,-0.098,-0.094,-0.091,-0.089,-0..086,-0.083,
-0.081,-0.079,-0.077,-0.075,-0.073,-0.072,-0.070,-0.069,-0.067
Alfa3 = 0.40 Lam2
0.2582,0.2580,0.2473,0.2273,0.2000,0.1690,0.1371,0.1060,0.0764,0.0485,
0.0224,0.0100,-0...397,-0.0136,-0.0248,-0.0462,-0.0662,-0.0850,-0.1027,
-0.1194,-0.1352,-0.1501,-0.1643,-0.1778,-0.1906,-0.2026,-0.2142,-0.2253,
-0.2359,-0.2459,-0.2558,-0.2650,-0.2741,-0.2827,-0.2908,-0.2988,-0.3064,
-0.3139,-0.3210
Alfa3 = 0.40 Lam3
0.0129,0.0430,0.648,0.767,0.782,0.718,0.609,0.0482,0.0351,0.0223,0.0103,
0.004597,-0.000182,-0.006204,-0.0113,-0.0209,-0.0297,-0.0379,-0.0455,
-0.0525,-0.0591,-0.0651,-0.0708,-0.0761,-0.0811,-0.0857,-0.0901,-0.0942,
-0.0981,-0.1018,-0.1053,-0.1086,-0.1118,-0.1148,-0.1176,-0.1203,-0.1229,
-0.1254,-0.1278
Alfa3 = 0.40 Lam4
0.5683,0.4500,0.3527,0.2720,0.2069,0.1555,0.1149,0.0824,0.0558,0.0337,
0.0149,0.006521,-0.000254,-0.008533,-0.0153,-0.0277,-0.0387,-0.0485,
-0.0574,-0.0654,-0.0727,-0.0794,-0.0856,-0.0913,-0.0966,-0.1014,-0.1060,
-0.1103,-0.1143,-0.1180,-0.1216,-0.1249,-0.1281,-0.1311,-0.139,-0.1366,
-0.1391,-0.1416,-0.1439
Alfa3 = 0.45 Lam1
-1.471,-1.138,-0.894,-0.707,--0.565,-0.460,-0.384,-0.329,-0.287,-0.255,
-0.230,-0.221,-0.208,-0.200,-0.192,-0.178,-0.165,-0.157,-0.147,-0.140,
-0.132,-0.127,-0.121,-0.116,-0.112,-0.108,-0.104,-0.101,-0.097,-0.095,
-0.092,-0.090,-0.088,-0.085,-0.084,-0.081,-0.080,-0.078,-0.076
Alfa3 = 0.45 Lam2
0.2500,0.2511,0.2424,0.2248,0.2003,0.1716,0.1412,0.1110,0.0818,0.0543,
0.0282,0.0158,0.004102,-0.007861,-0.0191,-0.0406,-0.0607,-0.0796,
-0.09765,-0.1142,-0.1302,-0.1453,-0.1595,-0.1731,-0.1860,-0.1983,
-0.2098,-0.2211,-0.2316,-0.2419,-0.2518,-0.2611,-0.2702,-0.2789,-0.2871,
-0.2952,-0.3029,-0.3102,-0.3176
Alfa3 = 0.45 Lam3

0.0000,0.0305,0.0528,0.0663,0.0707,0.0674,0.0590,0.480,0.361,0.241,0.0126
,0.007405,0.001833,-0.003505,-0.008511,-0.0180,-0.0268,-0.0349,-0.0425,
-0.0495,-0.0561,-0.0622,-0.0679,-0.0733,-0.0783,-0.0830,-0.0874,-0.0916,
-0.0955,-0.0992,-0.1028,-0.1061,-0.1093,-0.1124,-0.1152,-0.1180,-0.1206,
-0.1231,-0.1256
Alfa3 = 0.45 Lam4
0.5812,0.4608,0.3641,0.2840,0.2184,0.1657,0.1236,0.0897,0.0619,0.0388,
0.0193,0.0106,0.002691,-0.005065,-0.0121,+0.0249,-0.0362,-0.0464,
-0.0555,-0.0637,-0.0712,-0.0781,-0.0844,-0.0902,-0.1006,-0.1052,-0.1096,
-0.1136,-0.1175,-0.1211,-0.1245,-0.1277,-0.1307,-0.1336,-0.0363,-0.1389,
-0.1413,-0.1437
Alfa3 = 0.50 Lam1
-1.245,-0.987,-0.790,-0.639,-0.525,-0.440,-0.376,-0.329,-0.290,-0.262,
-0.248,-0.238,-0.228,-0.219,-0.202,-0.188,-0.177,-0.167,-0.157,-0.150,
-0.142,-0.137,-0.131,-0.126,-0.122,-0.117,-0.114,-0.110,-0.107,-0.104,
-0.101,-0.098,-0.095,-0.094,-0.091,-0.089,-0.088,-0.086,-0.084
Alfa3 = 0.50 Lam2
0.2445,0.2376,0.2225,0.2006,0.1742,0.1454,0.1163,0.0877,0.0604,0.0345,
0.0221,0.101,-0.001612,-0.0128,0.0344,0.0546,0.0737,0.0917,-0.1087,
-01246,-0.1398,-0.1542,-0.1679,-0.1809,0.1933,0.2050,-0.2163,-0.2270,
-0.2374,-0.2473,-0.2567,-0.2659,-0.2745,-0.2830,-0.2910,-0.2986,-0.3064,
-0.3134,-0.3206
Alfa3 = 0.50 Lam3
0.0178,0.0410,0.0561,0.0630,0.0625,0.0566,0.0476,0.0369,0.0259,0.0149,
0.009582,0.004383,0.-0.00700,-0.00570,-0.0149,-0.0236,-0.0317,-0.0393,
-0.0464,-0.0529,-0.0591,-0.0648,-0.0702,-0.0753,-0.0800,-0.0845,-0.0887,
-0.0927,-0.0965,-0.1001,-0.1035,-0.1067,-0.1098,-0.1127,-0.1155,-0.1181,
-0.1207,-0.1231,-0.1255
Alfa3 = 0.50 Lam4
0.4788,0.3770,0.2969,0.2307,0.1768,0.1332,0.0979,0.0689,0.0447,0.0243,
0.0152,0.006815,-0.001066,-0.008334,-0.0216,-0.0333,-0.0438,-0.0532,
-0.617,-0.0694,-0.0764,-0.0829,-0.0889,-0.0944,-0.0995,-0.1042,-0.1087,
-0.1128,-0.1167,-0.1204,-0.1238,-0.1271,-0.1301,-0.1331,-0.1358,-0.1384,
-0.1410,-0.1433,-0.1456
Alfa3 = 0.55 Lam1
-1.370,-1.087,-0.878,-0.716,-0.593,-0.499,-0.428,-0.372,-0.330,-0.298,
-0.269,-0.257,-0.247,-0.237,-0227,-0.213,-0.200,-0.187,-0.177,-0.169,
-0.161,-0.153,-0.147,-0.141,-0.136,-0.131,-0.127,-0.123,-0.119,-0.115,
-0.113,-0.110,-0.107,-0.104,-0.102,-0.100,-0.097,-0.095,-0.094
Alfa3 = 0.55 Lam2
0.2379,0.2331,0.2202,0.2009,0.1767,0.1497,0.1217,0.0940,0.0670,0.0413,
0.0170,0.005355,-0.005934,-0.0169,-0.0276,-0.0480,-0.0671,-0.0852,
-0.1024,-0.1184,-0.1338,-0.1483,-0.1620,-0.1753,-0.1878,-0.1197,-0.2111,
-0.2218,-0.2322,-0.2422,-0.2519,-0.2610,-0.2698,-0.2784,-0.2864,-0.2943,
-0.3019,-0.3092,-0.3164
Alfa3 = 0.55 Lam3
0.004463,0.0292,0.0459,0.0551,0.0572,0.0538,0.0467,0.0376,0.0275,0.0172,
0.007149,0.002258,-0.002515,-0.007160,-0.0117,-0.0203,-0.0283,-0.0359,
-0.0430,-0.0495,-0.0557,-0.0615,-0.0669,-0.0721,-0.0769,-0.0814,-0.0857,
-0.0897,-0.0935,-0.0972,-0.1006,-0.1039,-0.1070,-0.1100,-0.1128,-0.1155,
-0.1181,-0.1206,-0.1230
Alfa3 = 0.55 Lam4
0.4931,0.3920,0.3109,0.2440,0.1889,0.1438,0.1070,0.0767,0.0514,0.0301,
0.0118,0.003644,-0.003975,-0.0111,-0.0178,-0.0300,-0.0408,-0.0505,

-0.0593,-0.0672,-0.0745,-0.0811,-0.0872,-0.0929,-0.0981,0.1030,-0.1075,
-0.1117,-0.1157,-0.1194,-0.1230,-0.1263,-0.1294,-0.1324,-0.1352,-0.1379,
-0.1404,-0.1428,-0.1452
Alfa3 = 0.60 Lam1
-1.411,-1.198,-0.972,-0.800,-0.665,-0.562,-0.482,-0.420,-0.372,-0.335,
-0.302,-0.289,-0.277,-0.266,-0.256,-0.238,-0.222,-0.209,-0.197,-0.187,
-0.179,-0.171,-0.163,-0.157,-0.151,-0.146,-0.141,-0.137,-0.132,-0.128,
-0.124,-0.121,-0.118,-0.115,-0.113,-0.110,-0.108,-0.105,-0.103
Alfa3 = 0.60 Lam2
0.2347,0.2286,0.2180,0.2009,0.1791,0.1539,0.1273,0.1005,0.0740,0.0486,
0.0244,0.0128,0.001492,-0.009531,-0.0202,-0.0407,-0.0600,-0.0782,
-0.0956,-0.1118,-0.1273,0.1419,-0.1559,-0.1691,-0.1818,-0.1938,-0.2052,
-0.2163,-0.2267,-0.2368,-0.2465,-0.2557,-0.2647,-0.2732,-0.2815,-0.2894,
-0.2970,-0.3045,-0.3116
Alfa3 = 0.60 Lam3
0.0000,0.0171,0.0355,0.0467,0.0514,0.0504,0.0454,0.0379,0.0289,0.0194,
0.009911,0.005215,0.000611,-0.003916,-0.008326,-0.0168,-0.0248,
-0.0323,-0.0394,-0.0460,-0.0522,-0.0580,-0.0635,-0.0686,-0.0735,-0.0781,
-0.0824,-0.0865,-0.0904,-0.0941,-0.0976,-0.1009,-0.1041,-0.1071,-0.1100,
-0.1127,-0.1153,-0.1179,-0.1203
Alfa3 = 0.60 Lam4
0.4951,0.4098,0.3265,0.2583,0.2020,0.1554,0.1171,0.0854,0.0589,0.0366,
0.0175,0.008965,0.001025,-0.006425,-0.0134,-0.0261,-0.0373,-0.0474,
-0.0565,-0.0647,-0.0722,-0.0790,-0.0853,-0.0911,-0.0965,-0.1015,-0.1061,
-0.1105,-0.1145,-0.1183,-0.1219,-0.1253,-0.1285,-0.1315,-0.1344,-0.1371,
-0.1397,-0.1422,-0.1445
Alfa3 = 0.65 Lam1
-1.329,-1.076,-0.889,-0.744,-0.630,-0.542,-0.472,-0.418,-0.374,-0.338 ,
-0.324,-0.310,-0.297,-0.285,-0.265,-0.248,-0.231,-0.219,-0.209,-0.198,
-0.189,-0.181,-0.174,-0.167,-0.161,-0.155,-0.150,-0.145,-0.141,-0.137,
-0.134,-0.130,-0.127,-0.124,-0.121,-0.119,-0.116,-0.114,-0.112
Alfa3 = 0.65 Lam2
0.2240,0.2157,0.2010,0.1812,0.1582,0.1330,0.1072,0.0813,0.0564,0.0323,
0.0207,0.009399,-0.001593,-0.0123,-0.0328,-0.0524,-0.0707,-0.0880,
-0.1046,-0.1201,-0.1350,-0.1491,-0.1625,-0.1753,-0.1874,-0.1991,-0.2100,
-0.2208,-0.2309,-0.2407,-0.2501,-0.2591,-0.2677,-0.2761,-0.2840,-0.2919,
-0.2994,-0.3065,-0.3136
Alfa3 = 0.65 Lam3
0.003908,0.0246,0.0380,0.0449,0.0464,0.0435,0.0377,0.0300,0.0215,0.0126,
0.008137,0.003719,-0.000634,-0.004921,-0.0132,-0.0211,-0.0286,-0.0356,
-0.0422,-0.0484,-0.0543,-0.0598,-0.0650,-0.0700,-0.0746,-0.0790,-0.0831,
-0.0871,-0.0908,-0.0944,-0.0977,-0.1010,-0.1040,-0.1069,-0.1097,-0.1124,
-0.1150,-0.1174,-0.1198
Alfa3 = 0.65 Lam4
0.4318,0.3443,0.2742,0.2162,0.1682,0.1283,0.0952,0.0674,0.0440,0.0239,0.0
150,0.006660,-0.001106,-0.008391,-0.0216,-0.0334,-0.0438,-0.0532,
-0.0618,-0.0695,-0.0766,-0.0831,-0.0891,-0.0946,-0.0997,-0.1045,-0.1089,
-0.1131,-0.1170,-0.1207,-0.1242,-0.1274,-0.1305,-0.1335,-0.1362,-0.1389,
-0.1414,-0.1438,-0.1461
Alfa3 = 0.70 Lam1
-1.368,-1.194,-0.987,-0.828,-0.704,-0.606,-0.529,-0.467,-0.419,-0.379,
-0.344,-0.331,-0.317,-0.305,-0.294,-0.276,-0.257,-0.243,-0.229,-0.219,
-0.209,-0.199,-0.191,-0.184,-0.177,-0.170,-0.165,-0.160,-0.155,-0.161,
-0.147,-0.143,-0.139,-0.136,-0.133,-0.130,-0.127,-0.125,-0.122

Alfa3 = 0.70 Lam2
0.2217,0.2132,0.2008,0.1833,0.1621,0.1385,0.1139,0.0889,0.0643,0.0406,
0.0178,0.006799,-0.003917,-0.0144,-0.0245,-0.0441,-0.0626,-0.0802,
-0.0967,-0.1125,-0.1275,-0.1417,-0.1554,-0.1682,-0.1805,-0.1923,-0.2036,
-0.2144,-0.2246,-0.2346,-0.2439,-0.2532,-0.2618,-0.2703,-0.2784,-0.2862,
-0.2937,-0.3011,-0.3081
Alfa3 = 0.70 Lam3
0.0000,0.0130,0.0286,0.0378,0.0416,0.0409,0.0369,0.0307,0.0232,0.0151,
0.006767,0.002607,-0.001512,-0.005574,-0.009565,-0.0173,-0.0247,
-0.0317,-0.0383,-0.0445,-0.0504,-0.0560,-0.0613,-0.0662,-0.0709,-0.0745,
-0.0796,-0.0836,-0.0874,-0.0910,-0.0944,-0.0977,-0.1008,-0.1038,-0.1066,
-0.1093,-0.1119,-0.1144,-0.1168
Alfa3 = 0.70 Lam4
0.4353,0.3651,0.2918,0.2319,0.1821,0.1406,0.1060,0.0768,0.0522,0.0312,
0.0130,0.004872,-0.002750,-0.0099893,-0.0166,-0.0289,-0.0398,-0.0496,
-0.0584,-0.0665,-0.0738,-0.0805,-0.0867,-0.0924,-0.0977,-0.1026,-0.1072,
-0.1115,-0.1155,-0.1193,-0.1228,-0.1262,-0.1293,-0.1323,-0.1352,-0.1379,
-0.1404,-0.1429,-0.1452
Alfa3 = 0.75 Lam1
-1.334,-1.097,-0.921,-0.785,-0.677,-0.590,-0.521,-0.466,-0.419,-0.384,
-0.367,-0.352,-0.339,-0.3224,-0.306,-0.284,-0.268,-0.254,-0.240,-0.229,
-0.219,-0.209,-0.201,-0.194,-0.188,-0.181,-0.175,-0.170,-0.165,-0.160,
-0.156,-0.152,-0.148,-0.145,-0.142,-0.138,-.135,-0.133,-0.130
Alfa3 = 0.75 Lam2
0.2104,0.2003,0.1850,0.1658,0.1440,0.1206,0.0966,0.0726,0.0482,0.0266,
0.0156,0.004940,-0.005509,-0.0157,-0.0353,-0.0539,-0.0716,-0.0884,
-0.1044,-0.1195,-0.1339,-0.1476,-0.1607,-0.1731,-0.1851,-0.1964,-0.2074,
-0.2177,-0.2278,-0.2375,-0.2466,-0.2554,-0.2640,-0.2722,-0.2802,-0.2879,
-0.2952,-0.3023,-0.3093
Alfa3 = 0.75 Lam3
0.000,0.0183,0.0299,0.0360,0.0375,0.0355,0.0309,0.0246,0.0174,0.009663,
0.005749,0.001833,-0.002061,-0.005915,-0.0134,-0.0207,-0.0276,-0.0342,
-0.0405,-0.0464,-0.0520,-0.0573,-0.0623,-0.0670,-0.0715,-0.0758,-0.0799,
-0.0837,-0.0874,-0.0909,-0.0942,-0.0974,-0.1004,-0.1033,-0.1061,-0.1088,
-0.1113,-0.1137,-0.1161
Alfa3 = 0.75 Lam4
0.3903,0.3119,0.2492,0.1974,0.1542,0.1179,0.0873,0.0614,0.0392,0.0202,
0.016,0.003583,-0.003916,-0.109,-0.0238,-0.0352,-0.0454,-0.0547,-0.0630,
-0.0706,-0.0776,-0.0840,-0.0899,-0.0954,-0.1005,-0.1052,-0.1096,-0.1137,
-0.1176,-0.1213,-0.1247,-0.1279,-0.1310,-0.1339,-0.1367,-0.1393,-0.1418,
-0.1442,-0.1465
Alfa3 = 0.80 Lam1
-1.225,-1.025,-0.874,-0.754,-0.657,-0.528,-0.519,-0.468,-0.425,-0.392,
-0.375,-0.361,-0.349,-0.335,-0.313,-0.295,-0.279,-0.264,-0.251,-0.240,
-0.230,-0.220,-0.212,-0.204,-0.197,-0.191,-0.185,-0.180,-0.174,-0.169,
-0.166,-0.161,-0.157,-0.154,-0.150,-0.147,-0.144,-0.141,-0.139
Alfa3 = 0.80 Lam2
0.1996,0.1864,0.1692,0.1492,0.1272,0.1042,0.0810,0.0580,0.0357,0.0142,
0.003770,-0.006291,-0.0164,-0.0261,-0.0449,-0.0626,-0.0795,-0.0958,
-0.1110,-0.1255,-0.1394,-0.1527,-0.1653,-0.1774,-0.1889,-0.2000,-0.2104,
-0.2205,-0.2304,-0.2397,-0.2488,-0.2574,-0.2658,-0.2737,-0.2815,-0.2890,
-0.2962,-0.3033,-0.3100
Alfa3 = 0.80 Lam3
0.006847,0.0211,0.0295,0.0333,0.0333,0.0303,0.254,0.192,0.123,0.005035,

0.001352,-0.002278,-0.005981,-0.009598,-0.0167,-0.0235,-0.0300,-0.0363,
-0.0422,-0.0478,-0.0531,-0.0582,-0.0630,-0.0676,-0.0719,-0.0760,-0.0799,
-0.0836,-0.0872,-0.0906,-0.0938,-0.0969,-0.0999,-0.1027,-0.1054,-0.1080,
-0.1105,-0.1129,-0.1152
Alfa3 = 0.80 Lam4
0.3356,0.2687,0.2143,0.1691,0.1310,0.0989,0.0716,0.0482,0.0281,0.0107,
0.002770,-0.004531,-0.0116,-0.0181,-0.0301,-0.0408,-0.0504,-0.0592,
-0.0671,-0.0743,-0.0810,-0.0871,-0.0928,-0.0980,-0.1029,-0.1075,-0.1117,
-0.1157,-0.1195,-0.1230,-0.1264,-0.1295,-0.1325,-0.1353,-0.1380,-0.1406,
-0.1430,-0.1454,-0.1476
Alfa3 = 0.85 Lam1
-1.303,-1.145,-0.973,-0.838,-0.732,-0.645,-0.577,-0.519,-0.472,-0.430,
-0.413,-0.398,-0.383,-0.370,-0.344,-0.324,-0.305,-0.290,-0.275,-0.262,
-0.251,-0.241,-0.231,-0.223,-0.215,-0.207,-0.201,-0.195,-0.190,-0.184,
-0.179,-0.175,-0.171,-0.167,-0.163,-0.159,-0.156,0.153,-0.150
Alfa3 = 0.85 Lam2
0.1985,0.1875,0.1723,0.1541,0.1336,0.1119,0.0895,0.0671,0.0451,0.0238,
0.0134,0.003503,-0.006701,0.0165,-0.0353,-0.0531,-0.0703,-0.0864,
-0.1019,-0.1168,-0.1307,-0.1442,-0.1570,-0.1692,-0.1809,-0.1921,-0.2028,
-0.2130,-0.2229,-0.2324,-0.2416,-0.4503,-0.2587,-0.2669,-0.2748,-0.2823,
-0.2897,-0.2967,-0.3037
Alfa3 = 0.85 Lam3
0.0000,0.0110,0.0220,0.0281,0.0301,0.0291,0.0256,0.0206,0.0146,0.008001,
0.004581,0.001211,-0.002345,-0.005808,-0.0127,-0.0193,-0.0258,-0.0319,
-0.0378,-0.0435,-0.0488,-0.0539,-0.0588,-0.0634,-0.0678,-0.0720,-0.0759,
-0.0797,-0.0833,-0.0868,-0.0901,-0.0932,-0.0962,-0.0991,-0.1019,-0.1045,
-0.1071,-0.1095,-0.1119
Alfa3 = 0.85 Lam4
0.3488,0.2912,0.2332,0.1855,0.1455,0.1117,0.0829,0.0582,0.0370,0.0185,
0.0102,0.002612,-0.004896,-0.0118,-0.0244,-0.0356,-0.0457,-0.0548,
-0.0631,-0.0707,-0.0776,-0.0840,-0.0899,-0.0953,-0.1004,-0.1051,-0.1095,
-0.1136,-0.1175,-0.1211,-0.1246,-0.1278,-0.1309,-0.1338,-0.1366,-0.1392,
-0.1417,-0.1441,-0.1464
Alfa3 = 0.90 Lam1
-1.277,-1.085,-0.933,-0.814,-0.717,-0.639,-0.575,-0.522,-0.478,-0.439,
-0.422,-0.407,-0.394,-0.379,-0.353,-0.334,-0.317,-0.301,-0.287,-0.273,
-0.262,-0.252,-0.242,-0.233,-0.225,-0.218,-0.212,-0.205,-0.199,-0.194,
-0.189,-0.185,-0.180,-0.176,-0.182,-0.168,-0.165,-0.162,-0.159
Alfa3 = 0.90 Lam2
0.1880,0.1751,0.1586,0.1397,0.1193,0.0979,0.0762,0.0547,0.0337,0.0132,
0.003339,-0.006388,-0.0159,-0.0252,-0.0432,-0.0605,-0.0768,-0.0924,
-0.1073,-0.1215,-0.1352,-0.1481,-0.1606,-0.1723,-0.1838,-0.1947,-0.2051,
-0.2151,0.2246,-0.2340,-0.2428,-0.2514,-0.2597,-0.2676,-0.2753,-0.2827,
-0.2900,-0.2969,-0.3035
Alfa3 = 0.90 Lam3
0.0000,0.0133,0.0218,0.0260,0.0251,0.0214,0.0164,0.0106,0.004328,
0.001111,-0.002154,-0.005428,-0.008694,-0.0152,-0.0215,-0.0275,-0.0334,
-0.0390,-0.0444,-0.0495,-0.0544,-0.0591,-0.0635,-0.0678,-0.0718,-0.0756,
-0.0793,-0.0828,-0.0862,-0.0894,-0.0924,-0.0954,-0.0982,-0.1009,-0.1035,
-0.1060,-0.1084,-0.1107
Alfa3 = 0.90 Lam4
0.3160,0.2548,0.2039,0.1615,0.1258,0.0953,0.0693,0.0468,0.0273,0.0102,
0.002526,-0.004735,-0.0116,-0.0180,-0.0298,-0.0405,-0.0500,-0.0587,
-0.0666,-0.0738,-0.0805,-0.0866,-0.0923,-0.0975,-0.1024,-0.1070,-0.1113,

-0.1153,-0.1190,-0.1226,-0.1259,-0.1291,-0.1321,-0.1349,-0.1376,-0.1402,
-0.1427,-0.1450,-0.1472
Alfa3 = 1.00 Lam1
-1.253,-1.169,-1.010,-0.886,-0.787,-0.706,-0.638,-0.581,-0.533,-0.492,
-0.474,-0.445,-0.442,-0.429,-0.403,-0.379,-0.358,-0.341,-0.325,-0.309,
-0.297,-0.285,-0.275,-0.265,-0.256,-0.248,-0.241,-0.233,-0.220,-0.215,
-0.210,-0.195,-0.191,-0.187,-0.184,-0.180
Alfa3 = 1.00 Lam2
0.1772,0.1664,0.1509,0.1333,0.1142,0.0943,0.0741,0.0539,0.0340,0.0146,
0.005192,-0.000317,-0.0132,-0.0222,-0.0395,-0.0562,-0.0721,-0.0873,
-0.1019,-0.1158,-0.1291,-0.1419,-0.1540,-0.1685,-0.1769,-0.1878,-0.1980,
-0.2079,-0.2174,-0.2267,-0.2356,-0.2440,-0.2522,-0.2602,-0.2678,-0.2752,
-0.2824,-0.2893,-0.2959
Alfa3 = 1.00 Lam3
0.000,0.004828,0.0141,0.0193,0.0212,0.0206,0.0182,0.0144,0.009695,
0.004383,0.001584,-0.000101,-0.004176,-0.007097,-0.0129,-0.0187,
-0.0244,0.0299,-0.0352,-0.0404,-0.0453,-0.0500,-0.0545,-0.0589,-0.0630,
-0.0670,-0.0707,-0.074,-0.0778,-0.0812,-0.0844,-0.0874,-0.0904,-0.0932,
-0.0959,-0.0985,-0.1010,-0.1034,-0.1057
Alfa3 = 1.00 Lam4
0.2854,0.2490,0.1996,0.1588,0.1244,0.0950,0.0697,0.0477,0.0285,0.0117,
0.004061,-0.000242,-0.009946,-0.0164,-0.0282,-0.0388,-0.0484,-0.0571,
-0.0651,-0.0723,-0.0790,-0.0852,-0.0909,-0.0962,-0.1011,-0.1058,-0.1101,
-0.1141,-0.1179,-0.1215,-0.1249,-0.1281,-0.1311,-0.1340,-0.1367,-0.1393,
-0.1481,-0.1442,-0.1464
Alfa3 = 1.10 Lam1
-1.215,-1.108,-0.974,-0.869,-0.781,-0.708,-0.647,-0.596,-0.552,-0.532,
-0.517,-0.497,-0.481,-0.451,-0.427,-0.403,-0.834,-0.366,-0.350,-0.335,
-0.322,-0.311,-0.299,-0.289,-0.280,-0.271,-0.263,-0.256,-0.249,-0.242,
-0.236,-0.231,-0.226,-0.221,-0.216,-0.211,-0.203,-0.199
Alfa3 = 1.10 Lam2
0.1582,0.1459,0.1294,0.1117,0.0932,0.0743,0.0552,0.0365,0.0181,0.009038,
0.000997,-0.008629,-0.0173,-0.0340,-0.0501,-0.0656,-0.0805,-0.0947,
-0.1084,-0.1214,-0.1341,-0.1460,-0.1577,-0.1687,-0.1794,-0.1896,-0.1994,
-0.2090,-0.2180,-0.2267,-0.2353,-0.2435,-0.2513,-0.2590,-0.2664,-0.2735,
-0.2804,-0.2870,-0.2936
Alfa3 = 1.10 Lam3
0.0000,0.006035,0.0125,0.0157,0.0165,0.0154,0.0128,0.009168,0.004839,
0.002484,0.000279,-0.002479,-0.005046,-0.0103,-0.0155,-0.0208,-0.0259,
-0.0309,-0.0358,-0.0405,-0.0451,-0.0494,-0.0537,-0.0577,-0.0616,-0.0653,
-0.0689,-0.0724,-0.0757,-0.0788,-0.0819,-0.0848,-0.0876,-0.0903,-0.0930,
-0.0955,-0.0979,-0.1002,-0.1025
Alfa3 = 1.10 Lam4
0.2379,0.2013,0.1607,0.1267,0.0977,0.0828,0.0508,0.0318,0.0150,0.007342,
0.000795,-0.006726,-0.0132,-0.0251,-0.0358,-0.0455,-0.0544,-0.0624,
-0.0698,-0.0766,-0.0829,-0.0887,-0.0941,-0.0991,-0.1038,-0.1082,-0.1123,
-0.1162,-0.1198,-0.1232,-0.1265,-0.1296,-0.1325,-0.1353,-0.1379,-0.1404,
-0.1428,-0.1451,-0.1473
Alfa3 = 1.20 Lam1
-1.183,-1.083,-0.965,-0.870,-0.792,-0.723,-0.668,-0.619,-0.597,-0.577,
-0.558,-0.532,-0.508,-0.481,-0.454,-0.432,-0.412,-0.394,-0.378,-0.362,
-0.349,-0.337,-0.325,-0.314,-0.305,-0.296,-0.287,-0.280,-0.273,-0.265,
-0.259,-0.254,-0.248,-0.242,-0.237,-0.233,-0.228,-0.224,-0.220
Alfa3 = 1.20 Lam2

0.1407,0.1278,0.1113,0.0941,0.0764,0.0586,0.0408,0.0233,0.0146,0.006088,
-0.002319,-0.000962,-0.0268,-0.0424,-0.0575,-0.0719,-0.0860,-0.0993,
-0.1123,-0.1247,-0.1366,-0.140,-0.1589,-0.1695,-0.1796,-0.1896,-0.1990,
-0.2082,-0.2168,-0.2253,-0.2335,-0.2414,-0.2490,-0.2564,-0.2636,-0.2704,
-0.2772,-0.2837,-0.2901
Alfa3 = 1.20 Lam3
0.0000,0.00596,0.00968,0.0122,0.0124,0.0112,0.008705,0.005411,0.003525,
0.001515,-0.000594,-0.000245,-0.007343,-0.0120,-0.0168,-0.0215,-0.0262,
-0.0308,-0.0353,-0.0397,-0.0439,-0.0480,-0.0519,-0.0558,-0.0594,-0.0630,
-0.0664,-0.0697,-0.0728,-0.0759,-0.0788,-0.0816,-0.0843,-0.0870,-0.0895,
-0.0919,-0.0943,-0.0966,-0.0988
Alfa3 = 1.20 Lam4
0.1997,0.1675,0.1329,0.1036,0.0784,0.0565,0.0372,0.0202,0.0124,0.005050,
-0.001884,-0.000784,-0.0206,-0.0315,-0.0414,-0.0504,-0.0587,-0.0662,
-0.0732,-0.0796,-0.0856,-0.0911,-0.0962,-0.1010,-0.1055,-0.1098,-0.1137,
-0.1175,-0.1210,-0.1243,-0.1275,-0.1305,-0.1333,-0.1360,-0.1386,-0.1410,
-0.1434,-0.1456,-0.1478
Alfa3 = 1.30 Lam1
-1.156,-1.084,-0.975,-0.886,-0.812,-0.749,-0.695,-0.604,-0.617,-0.616,
-0.489,-0.572,-0.539,-0.510,-0.485,-0.463,-0.442,-0.424,-0.407,-0.392,
-0.378,-0.365,-0.353,-0.342,-0.332,-0.322,-0.314,-0.305,-0.298,-0.391,
-0.284,-0.277,-0.272,-0.266,-0.261,-0.256,-0.251,-0.246,-0.242
Alfa3 = 1.30 Lam2
0.1244,0.1129,0.0968,0.0802,0.0634,0.0466,0.0300,0.000286,0.000446,
-0.000526,-0.0104,-0.0182,-0.0333,-0.0480,-0.0622,-0.0758,-0.0890,
-0.1017,-0.1140,-0.1258,-0.1372,-0.1480,-0.1584,-0.1687,-0.1784,-0.1878,
-0.1969,-0.2057,-0.2141,-0.2223,-0.2304,-0.2379,-0.2453,-0.2525,-0.2595,
-0.2662,-0.2728,-0.2792,-0.2852
Alfa3 = 1.30 Lam3
0.0000,0.003174,0.007225,0.009035,0.009148,0.007959,0.005783,0.6619$,
0.000100,-0.000118,-0.002450,-0.004399,-0.008469,-0.0127,-0.0170,
-0.0213,-0.0256,-0.0298,-0.0340,-0.0380,-0.0420,-0.0458,-0.0495,-0.0531,
-0.0566,-0.0600,-0.0632,-0.0664,-0.0694,-0.0723,-0.0752,-0.0779,-0.0805,
-0.0831,-0.0855,-0.0879,-0.0902,-0.0925,-0.0946
Alfa3 = 1.30 Lam4
0.1679,0.1435,0.1130,0.0870,0.0645,0.0447,0.027300.00239,0.00375,
-0.000442,-0.008504,-0.0146,-0.0258,-0.0360,-0.0453,-0.0538,-0.0616,
-0.0688,-0.0754,-0.0816,-0.0873,-0.0926,-0.0975,-0.1022,-0.1106,-0.1145,
-0.1181,-0.1215,-0.1248,-0.1279,-0.1308,-0.1336,-0.1362,-0.1388,-0.1412,
-0.1435,-0.1457,-0.1478
Alfa3 = 1.40 Lam1
-1.132,-1.106,-1.001,-0.916,-0.844,-0.782,-0.729,-0.706,-0.683,-0.660,
-0.643,-0.607,-0.575,-0.547,-0.521,-0.498,-0.475,-0.458,-0.440,-0.423,
-0.410,-0.395,-0.383,-0.372,-0.361,-0.351,-0.342,-0.333,-0.325,-0.317,
-0.310,-0.303,-0.397,-0.291,-0.285,-.279,-0.274,-0.269,-0.265
Alfa3 = 1.40 Lam2
0.1092,0.1011,0.0855,0.0697,0.0538,0.0379,0.0222,0.0145,0.006822,
-0.001226,-0.008266,-0.0230,-0.0373,-0.0510,-0.0645,-0.0775,-0.0900,
-0.1020,-0.1137,-0.1250,-0.1358,-0.1463,-0.1564,-0.1662,-0.1756,-0.1846,
-0.1935,-0.2018,-0.2102,-0.2181,-0.2257,-0.2332,-0.2405,-0.2475,-0.2542,
-0.2609,-0.2671,-0.2734,-0.2794
Alfa3 = 1.40 Lam3
0.0000,0.00787,0.004546,0.006396,0.006530,0.005603,0.003785,0.002311,
0.001292,-0.000244,-0.001702,-0.005060,-0.008670,-0.0124,-0.0163,

-0.0202,-0.0242,-0.0280,-0.0319,-0.0357,-0.0393,-0.0430,-0.0499,-0.0532,
-0.0564,-0.0595,-0.0625,-0.0655,-0.0683,-0.0710,-0.0737,-0.0762,-0.0787,
-0.0811,-0.0835,-0.0857,-0.0879,-0.0900
Alfa3 = 1.40 Lam4
0.1411,0.1268,0.0991,0.0754,0.0547,0.0365,0.0204,0.0130,0.005987,
-0.001052,-0.006968,-0.0187,-0.0293,-0.0389,-0.0478,-0.0559,-0.0633,
-0.0702,-0.0766,-0.0825,-0.0881,-0.0932,-0.0980,-0.1026,-0.1068,-0.1108,
-0.1146,-0.1181,-0.1215,-0..1247,-0.1277,-0.1306,-0.1334,-0.1360,-0.1385,
-0.1409,-0.1431,-0.1453,-0.1474
Alfa3 = 1.50 Lam1
-1.112,-1.103,-1.042,-0.957,-0.885,-0.824,-0.688,-0.747,-0.714,-0.704,
-0.684,-0.647,-0.615,-0.585,-0.558,-0.536,-0.514,-0.494,-0.476,-0.459,
-0.443,-0.429,-0.416,-0.404,-0.392,-0.382,-0.372,-0.363,-0.354,-0.346,
-0.338,-0.331,-0.325,-0.317,-0.311,-0.305,-0.300,-0.295,-0.289
Alfa3 = 1.50 Lam2
0.0951,0.0886,0.0773,0.0622,0.0471,0.0321,0.000566,0.009962,-0.000290,
-0.004446,-0.0115,-0.0254,-0.0390,-0.0520,-0.0648,-0.0767,-0.0891,
-0.1007,-0.1118,-0.1225,-0.1330,-0.1431,-0.1528,-0.1622,-0.1713,-0.1803,
-0.1887,-0.1969,-0.2049,-0.2127,-0.2202,-0.2273,-0.2339,-0.2414,-0.2478,
-0.2544,-0.2607,-0.2662,-0.2726
Alfa3 = 1.50 Lam3
0.0000,0.0000,0.001949,0.003907,0.004441,0.003885,0.000104,0.001538,
-0.4897$,-0.000768,-0.002088,-0.004989,-0.008156,-0.01155,-0.0150,
-0.0184,-0.0221,-0.0257,-0.0292,-0.0327,-0.0362,-0.0396,-0.0429,-0.0461,
-0.0493,-0.0524,-0.0553,-0.0582,-0.0611,-0.0638,-0.0665,-0.0690,-0.0713,
-0.0740,-0.0763,-0.0786,-0.0808,-0.0827,-0.0851
Alfa3 = 1.50 Lam1
0.1182,0.1083,0.0899,0.0677,0.0483,0.0313,0.000494,0.009059,-0.000256,
-0.003882,-0.009875,-0.0210,-0.0312,-0.0404,-0.0489,-0.0565,-0.0640,
-0.0707,-0.0769,-0.0826,-0.0880,-0.0931,-0.0978,-0.1022,-0.1064,-0.1104,
-0.1141,-0.1176,-0.1209,-0.1241,-0.1271,-0.1299,-0.1325,-0.1353,-0.1377,
-0.1401,-0.1424,-0.1444,-0.1466
Alfa3 = 1.60 Lam4
-1.086,-1.078,-1.011,-0.937,-0.875,-0.846,-0.796,-0.771,-0.751,-0.731,
-0.693,-0.659,-0.630,-0.602,-0.5778,-0.553,-0.534,-0.515,-0.496,-0.480,
-0.465,-0.452,-0.438,-0.426,-0.415,-0.404,-0.394,-0.385,-0.377,-0.368,
-0.360,-0.352,-0.346,-0.339,-0.333,-0.328,-0.321,-0.316,-0.311
Alfa3 = 1.60 Lam1
-1.086,-1.078,-1.011,-0.937,-0.875,-0.746,-0.796,-0.771,-0.751,-0.731,
-0.693,-0.659,-0.630,-0.602,-0.577,-0.553,-0.534,-0.515,-0.496,-0.480,
-0.465,-0.452,-0.438,-0.426,-0.415,-0.404,-0.394,-0.385,-0.377,-0.368,
-0.360,-0.352,-0.346,-0.339,-0.333,-0.328,-0.321,-0.316,-0.311
Alfa3 = 1.60 Lam2
0.0757,0.0698,0.0573,0.0430,0.0287,0.000422,0.0077738,-0.000341,
-0.005924,-0.0127,-0.0258,-0.0286,-0.0511,-0.0633,-0.0752,-0.0866,
-0.0972,-0.1084,-0.1187,-0.1288,-0.1385,-0.1480,-0.1572,-0.1659,-0.1745,
-0.1828,-0.1908,-0.1986,-0.2062,-0.2135,-0.2206,-0.2275,-0.2341,-0.2407,
-0.2471,-0.2527,-0.2592,-0.2650,-0.2706
Alfa3 = 1.60 Lam3
0.0000,0.0000,0.001699,0.002684,0.002597,0.6356$,0.000969,-0.4634$,
-0.000858,-0.001942,-0.004383,-0.007111,-0.0100,-0.0131,-0.0163,
-0.0196,-0.0227,-0.0261,-0.0294,-0.0326,-0.0358,-0.0289,-0.0420,-0.0450,
-0.0479,-0.0508,-0.0536,-0.0563,-0.0589,-0.0615,-0.0640,-0.0665,-0.0688,
-0.0711,-0.0734,-0.0753,-0.0777,-0.0797,-0.0817

Alfa3 = 1.60 Lam4
0.0896,0.0814,0.0634,0.0449,0.0285,0.000378,0.007177,-0.000309,
-0.005279,0.0111,-0.0218,-0.0316,-0.0406,-0.0489,-0.0566,-0.0636,
-0.0699,-0.0763,-0.0819,-0.0872,-0.0922,-0.0969,-0.1013,-0.1054,-0.1093,
-0.1130,-0.1165,-0.1198,-0.1230,-0.1260,-0.1288,-0.1315,-0.1341,-0.1366,
-0.1390,-0.1411,-0.1434,-0.1455,-0.1475
Alfa3 = 1.70 Lam1
-1.064,-1.057,-1.001,-0.935,-0.878,-0.852,-0.825,-0.806,-0.784,-0.745,
-0.709,-0.678,-0.650,-0.622,-0.598,-0.578,-0.557,-0.538,-0.521,-0.505,
-0.489,-0.476,-0.463,-0.451,-0.440,-0.429,-0.419,-0.410,-0.401,-0.392,
-0.384,-0.377,-0.369,-0.362,-0.356,-0.350,-0.344,-0.338,-0.333
Alfa3 = 1.70 Lam2
0.0580,0.0525,0.0412,0.0275,0.0142,0.007546,-0.000250,-0.005469,
-0.0119,-0.0245,-0.0367,-0.0487,-0.0603,-0.0717,-0.0827,-0.0933,-0.1036,
-0.1136,-0.1233,-0.1329,-0.120,-0.1509,-0.1594,-0.1677,-0.1758,-0.1837,
-0.1913,-0.1988,-0.2059,-0.2128,-0.2195,-0.2261,-0.2326,-0.2388,-0.2450,
-0.2508,-0.2566,-0.2622,-0.2675
Alfa3 = 1.70 Lam3
0.0000,0.0000,0.001027,0.001513,0.001142,0.000696,-0.2601$,-0.000619,
-0.001463,-0.003423,-0.005705,-0.008225,-0.0109,-0.0138,-0.0167,
-0.0196,-0.0226,-0.0256,-0.0286,-0.0316,-0.0346,-0.0375,-0.0403,-0.0431,
-0.0458,-0.0485,-0.0511,-0.0537,-0.0532,-0.0586,-0.0610,-0.0633,-0.0656,
-0.0678,-0.0700,-0.0720,-0.0741,-0.0761,-0.0780
Alfa3 = 1.70 Lam4
0.0657,0.0588,0.0441,0.0280,0.0138,0.007179,-0.000232,-0.005000,
-0.0107,-0.0212,-0.0308,-0.0397,-0.0478,-0.0553,-0.0623,-0.0688,-0.0748,
-0.0804,-0.0857,-0.0907,-0.0953,-0.0997,-0.1038,-0.1077,-0.01114,
-0.1149,-0.1182,-0.1214,-0.1244,-0.1272,-0.1299,-0.1325,-0.1350,-0.1374,
-0.1397,-0.1419,-0.1440,-0.1460,-0.1479
Alfa3 = 1.80 Lam1
-1.045,-1.039,-1.007,-0.945,-0.918,-0.892,-0.868,-0.846,-0.804,-0.767,
-0.733,-0.702,-0.675,-0.649,-0.625,-0.604,-0.583,-0.565,-0.548,-0.532,
-0.517,-0.503,-0.490,-0.478,-0.467,-0.456,-0.445,-0.436,-0.427,-0.456,
-0.445,-0.436,-0.427,-0.418,-0.410,-0.402,-0.395,-0.388,-0.381,-0.374,
-0.368,-0.362,-0.357
Alfa3 = 1.80 Lam2
0.0417,0.0367,0.024,0.0155,0.009177,0.002914,-0.003291,-0.009427,
-0.0215,-0.0333,-0.0448,-0.0559,-0.0668,-0.0774,-0.0877,-0.0978,-0.1075,
-0.1169,-0.1260,-0.1349,-0.1436,-0.1520,-0.1600,-0.1679,-0.1757,-0.1831,
-0.1904,-0.1974,-0.2043,-0.2109,-0.2175,-0.2238,-0.2299,-0.2359,-0.2417,
-0.2473,-0.2530,-0.2583,-0.2632
Alfa3 = 1.80 Lam3
0.0000,0.0000,0.000378,0.000646,0.000498,0.000193,-0.000254,
-0.000826,-0.002289,-0.004103,-0.006190,-0.008489,-0.0109,-0.0135,
-0.0162,-0.0189,-0.0217,-0.0244,-0.0272,-0.0299,-0.0327,-0.0354,-0.0380,
-0.0406,-0.0432,-0.0457,-0.0482,-0.0506,-0.0530,-0.0553,-0.0576,-0.0598,
-0.0619,-0.0640,-0.0661,-0.0681,-0.0701,-0.0720,-0.0737
Alfa3 = 1.80 Lam4
0.0456,0.0396,0.0298,0.0155,0.009006,0.002801,-0.003102,-0.008721,
-0.0192,-0.0288,-0.0376,-0.0456,0.0531,-0.0601,-0.0668,-0.0726,-0.0782,
-0.0835,-0.0884,-0.0931,-0.0975,-0.1016,-0.1055,-0.1092,-0.1128,-0.1161,
-0.1193,-0.1223,-0.1252,-0.1279,-0.1306,-0.1331,-0.1355,-0.1378,-0.1400,
-0.1421,-0.1442,-0.1461,-0.1479
Alfa3 = 1.90 Lam1

-1.023,-1.018,-0.968,-0.946,-0.917,-0.893,-0.871,-0.831,-0.794,-0.761,
-0.731,-0.703,-0.679,-0.656,-0.634,-0.614,-0.595,-0.578,-0.562,-0.547,
-0.533,-0.520,-0.508,-0.495,-0.485,-0.474,-0.464,-0.455,-0.446,-0.437,
-0.429,-0.421,-0.414,-0.407,-0.400,-0.394,-0.388,-0.385,-0.376
Alfa3 = 1.90 Lam2
0.0220,0.0175,0.006447,0.001239,-0.005444,-0.0113,-0.0171,-0.0284,
-0.0395,-.0503,-0.0609,-0.0712,-0.0811,-0.0907,-0.1002,-0.1093,-0.1183,
-0.1269,-0.1355,-1437,-0.1515,-0.1594,-0.1665,-0.1742,-0.1811,-0.1883,
-0.1950,-0.2015,-0.2080,-0.2142,-0.2203,-0.2262,-0.2320,-0.2376,-0.2431,
-0.2485,-0.2537,-0.2589,-0.2636
Alfa3 = 1.90 Lam3
0.0000,0.0000,0.000150,0.4120$,-0.000257,-0.000657,-0.001167,
-0.002475,-0.004100,-0.005975,-0.008046,-0.0103,-0.0126,-0.0150,
-0.0175,-0.0200,-0.0226,-0.0251,-0.0277,-0.0302,-0.0327,-0.0352,-0.0375,
-0.0401,-0.0423,-0.0448,-0.0471,-0.0493,-0.0515,-0.0537,-0.0558,-.0579,
-0.0599,-0.0619,-0.0638,-0.0657,-0.0676,-0.0694,-0.0711
Alfa3 = 1.90 Lam4
0.0230,0.0181,0.006431,0.001215,-0.005220,-0.0106,-0.0158,-0.0254,
-0.0343,-0.0424,-0.0500,-0.0570,-0.0635,-0.0695,-0.0752,-0.0805,-0.0855,
-0.0902,-0.0947,-0.0989,-0.1028,-0.1066,-0.1100,-0.1135,-0.1166,-0.1198,
-0.1227,-0.1255,-0.1282,-0.1307,-0.1332,-0.1355,-0.1378,-0.1399,-0.1420,
-0.1440,-0.1459,-0.1478,-0.1495
Alfa3 = 2.00 Lam1
-1.009,-1.004,-1.002,-0.993,-0.974,-0.950,-0.905,-0.865,-0.828,-0.796,
-0.766,-0.738,-0.713,-0.690,-0.670,-0.647,-0.629,-0.611,-0.595,-0.579,
-0.565,-0.557,-0.539,-0.527,-0.515,-0.504,-0.495,-0.485,-0.475,-0.466,
-0.458,-0.450,-0.443,-0.436,-0.428,-0.422,-0.415,-0.409,-0.403
Alfa3 = 2.00 Lam2
0.008397,0.004147,0.002061,-0.001081,-0.005675,-0.0113,-0.0222,
-0.0331,-0.0435,-0.0538,-0.0637,-0.0734,-0.0713,-0.0690,-0.0670,-0.0647,
-0.0629,-0.0611,-0.0595,-0.0579,-0.0565,-0.0557,-0.0539,-0.0527,-0.0515,
-0.0504,-0.0495,-0.0485,-0.0475,-0.0466,-0.0458,-0.0450,-0.0443,-0.0436,
-0.0428,-0.0422,-0.0415,-0.0409,-0.0403
Alfa3 = 2.00 Lam3
0.0000,0.0000,0.0001$,-0.0407$,-0.7075$,-0.000272,-0.001012,-0.002125,
-0.003537,-0.005187,-0.007027,-0.009016,-0.0111,-0.0133,-0.0154,
-0.0179,-0.0202,-0.0226,-0.0249,-0.0273,-0.0296,-0.0312,-0.0342,-0.0365,
-0.0388,-0.0410,-0.0431,-0.0453,-0.0474,-0.0495,-0.0515,-0.0535,-0.0554,
-0.0571,-0.0592,-0.0610,-0.0628,-0.0646,-0.0663
Alfa3 = 2.00 Lam4
0.008541,0.004182,0.002070-0.001076,-0.005567,-0.0109,-0.0207,
-0.0298,-0.0381,-0.0458,-0.0429,-0.0595,-0.0657,-0.0714,-0.0766,-0.0819,
-0.0867,-0.0912,-0.0955,-0.0995,-0.1033,-0.1062,-0.1104,-0.1137,-0.1168,
-0.1198,-0.1226,-0.1254,-0.1280,-0.1305,-0.1329,-0.1351,-0.1373,-0.1393,
-0.1415,-0.1435,-0.1453,-0.1472,-0.1489